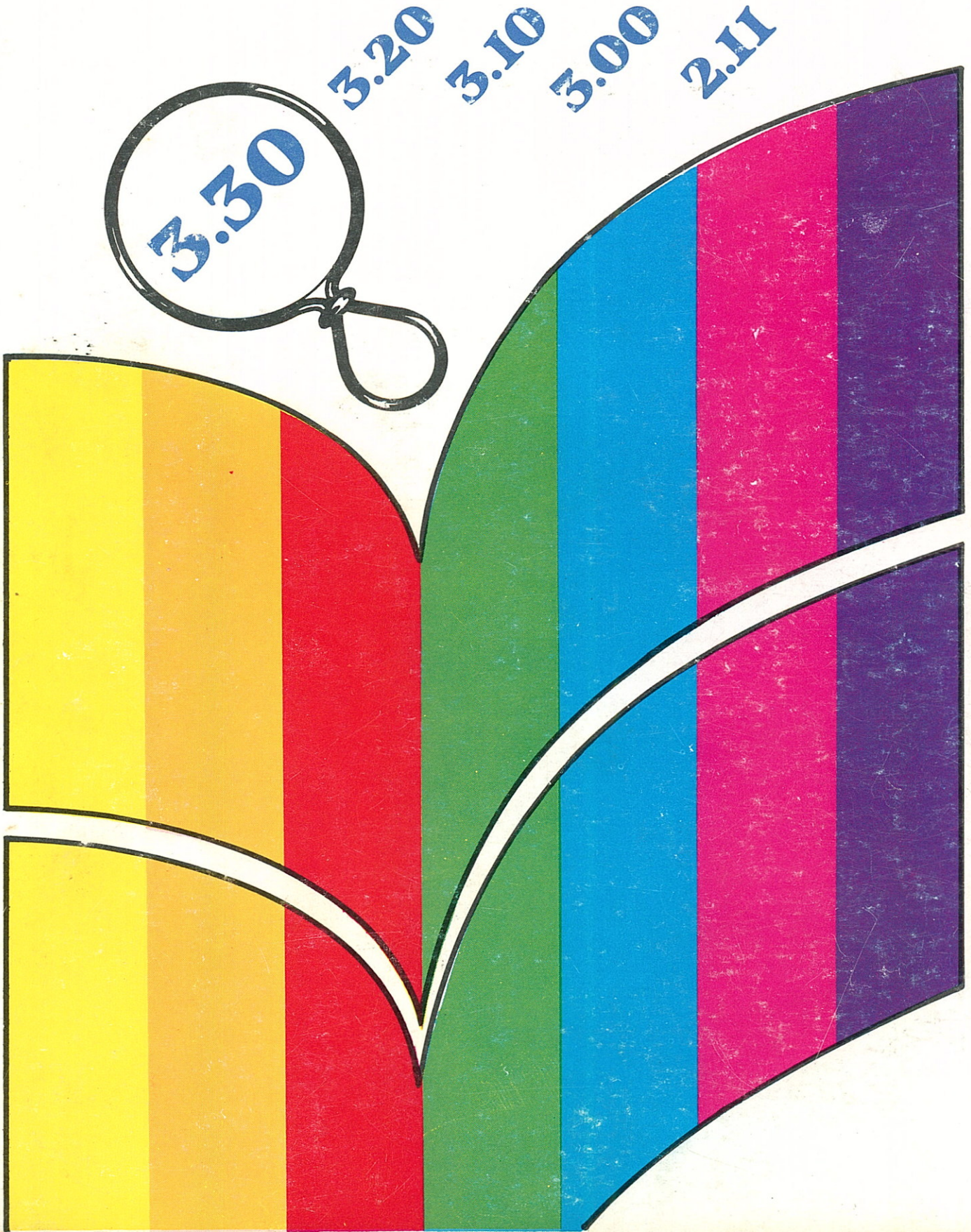
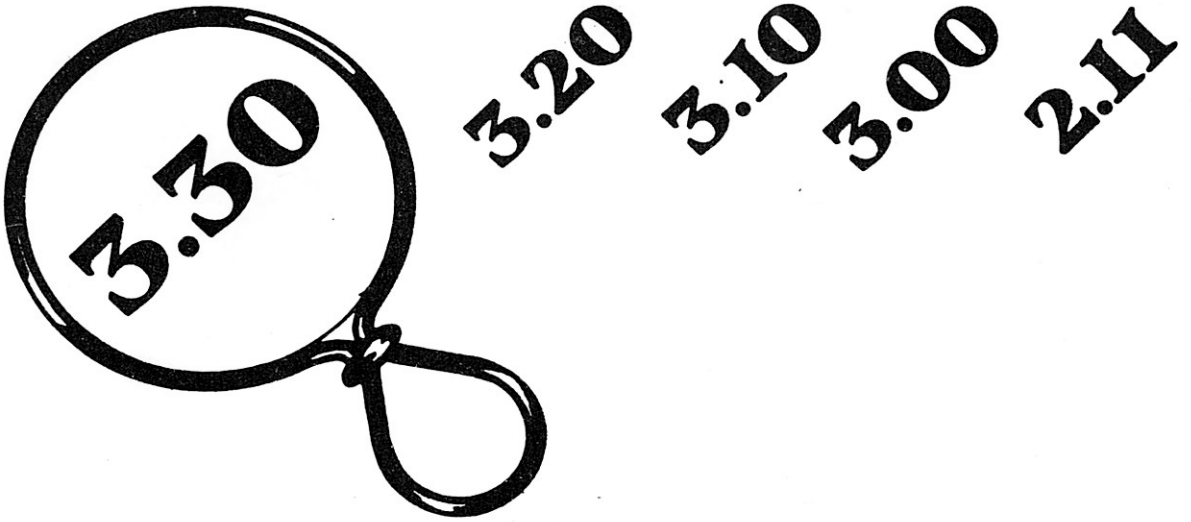
IBM PC DOS

II. A PC DOS FELÉPÍTÉSE



IBM PC DOS

II. A PC DOS FELÉPÍTÉSE



LSI
ALKALMAZÁSTECHNIKAI TANÁCSADÓ SZOLGÁLAT
BUDAPEST, 1988

Szerkesztette: Szenes Katalin I-II.
Dr. Úry László III.

Készítette: Bernáth István — 1, 2, 8 és 9. fejezet
Darvas László — 3, 4, 5, 6, 7, 11, 12, 13, 14, 15, 16 és 17. fejezet
Orosz Judit — 10. fejezet
Seres József — 3, 4, 5, 6, 7, 11, 12, 13, 14, 15, 16, és 17. fejezet
Úry László — 3.xx kiegészítések

III kötet: Orosz Judit
Baki Zoltán
Pajor Gábor

Lektorálta: Éltető László
Szenes Katalin
Darvas László
Donát János

Munkatárs: Sziklai Klára

Kiadó: LSI Alkalmazástechnikai Tanácsadó Szolgálat
Felelős kiadó: dr. Kovács Magda
Témafelelős: Székely László
Szedés: LSI ATSz Info Osztály
Technikai szerkesztő: Nagy Olivér
Engedélyszám: 57152
ISBN II. kötet: 963 592 812 2
Készült a Szabadság MgTsz Nyomdaüzemében, Gyál
Felelős vezető: Tóth Antal

TARTALOMJEGYZÉK

I. kötet

Előszó 3
Tartalomjegyzék 5

1. FEJEZET
Bevezetés 11

2. FEJEZET
PC DOS alapismeretek 17

3. FEJEZET
DOS: file-kezelő parancsok 41

4. FEJEZET
DOS: lemezkezelési parancsok 60

5. FEJEZET
A fix lemez kezelése 81

6. FEJEZET
A rendszer működését befolyásoló parancsok és kapcsolók 90

7. FEJEZET
A parancs-file-ok 105

8. FEJEZET
A sorszerkesztő program (EDLIN) 115

9. FEJEZET
A nyomkövető-hibakereső program (DEBUG) 141

10. FEJEZET
Futtatható program előállítása - programszerkesztés 161

11. FEJEZET
A rendszer konfigurálása 171

12. FEJEZET
A PC DOS speciális lehetőségei 184

II. kötet

13. FEJEZET

A PC DOS belső struktúrája 202
13.1 A parancsfeldolgozó program (COMMAND.COM) 202
13.2 A PC DOS indítása 203
13.3 A PC DOS-nak a felhasználói programok által elérhető szolgáltatásai, függvényhívási és megszakítási rendszere 205
13.3.1 A 8086-os processzor regiszterei 205
13.3.2 A 8086-os processzor címképzési rendszere 206
13.3.3 A megszakítás fogalma 207
13.3.4 A PC DOS megszakításainak alkalmazása felhasználói programban 210
13.3.5 A funkcióhívási rendszer 211
13.4 A DOS perifériakezelése 211
13.4.1 Karakteres berendezések 211
13.4.2 Blokkos berendezések 212

14. FEJEZET

Lemezkezelés és a felhasználói perifériakezelők szerkezete 213
14.1 A floppy és a winchester lemezek felépítése 213
14.1.1 A boot szektor és a partíciós tábla 214
14.1.2 A file leíró tábla (FAT) 217
14.1.3 A tartalomjegyzék (directory) felépítése, altartalomjegyzékek 220
14.1.4 Kiterjesztett DOS partíció felépítése 222
14.2 A felhasználói periferiakezelő program (handler, driver) 223
14.2.1 A perifériakezelő program felépítése 223
14.2.2 A perifériakezelő programokkal kapcsolatos tudnivalók összefoglalása és kiegészítése 225
14.2.3 A parancsblokk és fejlécének felépítése 227
14.2.4 A perifériakezelőt vezérlő parancsok részletes leírása 228
14.2.5 Javaslat gyakorlati példára 234

15. FEJEZET

A DOS memóriakezelése, munkaterületei és vezérlőblokkjai 236
15.1 A DOS memóriatérképe 236
15.2 A programszegmens prefix (PSP) 237
15.2.1 A PSP felépítése 238
15.2.2 A PSP egy program indításakor 239
15.3 A programkörnyezet es a memórianyilvántartás 240
15.3.1 Programkörnyezet 240
15.3.2 A memórianyilvántartás 241
15.4 A file-kezelés és memóriaterületeinek szerkezete 243
15.4.1 File-leíró blokk (FCB) 244
15.4.2 Lemezátviteli terület (DTA) 246
15.4.3 File-kezelés az FCB segítségével 246
15.4.4 A DOS file-szám (handle) és használata 250

16. FEJEZET

DOS megszakítások 252

16.1 A 20-as megszakítás: program befejezése 252
16.2 A 21-es megszakítás: DOS funkció meghívása 252
16.3 A 22-es megszakítás: program kilépési cím 252
16.4 A 23-as megszakítás: Ctrl-Break kilépési cím 253
16.5 A 24-es megszakítás: kritikus hibák kezelése 253
16.6 A 25-ös megszakítás: közvetlen lemezolvasás 257
16.7 A 26-os megszakítás: fizikai szintű lemezírás 257
16.8 A 27-es megszakítás: befejezés, rezidens programok részére 258
16.9 A 2F-es megszakítás: multiplex megszakítás 258

17. FEJEZET

DOS funkciók 260

I/O műveletek karakteres berendezésekkel 260
Processzor-kezelő funkciók 260
Memória kezelés 260
Tartalomjegyzékkel kapcsolatos műveletek 260
FCB-s file-kezelő funkciók 261
File-számos file-kezelés 261
Egyéb rendszerfunkciók 261
File megosztás és hálózatkezelés 262

17.1 A DOS funkciók hibakezelése 263
17.2 A DOS funkcióhívások általános szabályai 264
17.3 A DOS funkciók ismertetése 266

Funkció ..... **oldal**
"0" - program befejezése 266
"1" - tasztatúra input 267
"2" - kiírás képernyőre 267
"3" - soros vonali input 268
"4" - soros vonali output 268
"5" - karakter kiírása nyomtatóra 268
"6" - közvetlen konzol I/O 269
"7" - közvetlen konzol input képernyőre írással 269
"8" - konzol input képernyőre írás nélkül 270
"9" - karaktersorozat nyomtatása 270
"A" - pufferelt tasztatúra input 271
"B" - standard input egység állapotának ell. 271
"C" - a standard input puffer törlése, és egy standard input kezelő funkció hívása... 271
"D" - lemez visszaállítása 272
"E" - lemezegység kiválasztása 272
"F" - file megnyitása 273

"10" - file lezárása 273
"11" - adott FCB-beli file-név keresése tartalomjegyzékben 274
"12" - 11-es funkcióval megtalált file-t követő további file keresése 275
"13" - file törlése 275
"14" - file soros olvasása 276
"15" - file soros írása 276
"16" - file létrehozása 277
"17" - file átnevezése 277
"18" - a DOS számára FENNTARTOTT funkció 278
"19" - aktuális meghajtó kódjának visszanyerése 278
"1A" - a DTA címének beállítása 278
"1B" - aktuális meghajtó kódjának visszanyerése 278
"1C" - adott meghajtó FAT információina visszanyerése 279
"1D-20" - a DOS számára FENNTAROTT funkciók 279
"21" - random olvasás 279
"22" - random írás 280
"23" - file méret lekérdezése 280
"24" - a random rekord mező beállítása 281
"25" - a megszakítási (interrupt) vektor beállítása 281
"26" - új program szegmens előállítása 282
"27" - random blokk olvasása 282
"28" - random blokk írása 283
"29" - file azonosító keresése adott parancs-sorban 283
"2A" - dátum lekérdezése 285
"2B" - dátum beállítása 285
"2C" - rendszeridő lekérdezése 285
"2D" - rendszeridő beállítása 286
"2E" - VERIFY (lemezírás utáni ellenőrzés) kapcsoló beállítása 286
"2F" - lemez átviteli terület (DTA) címének lekérdezése 287
"30" - a DOS verziószám lekérdezése 287
"31" - programfutás befejezése úgy, hogy a program a memóriában marad 288
"32" - a DOS számára FENNTARTOTT funkció 288
"33" - a Ctrl-Break lekérdezése és beállítása 288
"34" - a DOS számára FENNTARTOTT funkció 289
"35" - megszakítási vektor lekérdezése 289
"36" - szabad lemezterület lekérdezése 289
"37" - a DOS számára FENNTARTOTT funkció 290
"38" - ország-specifikus információk lekérdezése 290
"39" - altartalomjegyzék létrehozása 291
"3A" - altartalomjegyzék törlése 292
"3B" - az aktuális tartalomjegyzék beállítása 292
"3C" - file létrehozása vagy meglévő file hosszának 0-ra állítása 293
"3D" - file megnyitása 293
"3E" - file lezárása 295
"3F" - olvasás file-ból, vagy egyéb perifériáról 295
"40" - írás file-ba vagy egyéb perifériára 296

"41" - file törlése adott tartalomjegyzékből 296
"42" - az írási/olvasási mutató átállítása 297
"43" - file attributumának lekérdezése vagy átállítása 298
"44" - periféria vagy file-leíró információ lekérdezése vagy átállítása 299
"45" - egy file számának duplikálása 302
"46" - file-szám duplikálás úgy, hogy a másodpéldányt a felhasználó adja meg 302
"47" - adott meghajtó aktuális tartalomjegyzékének lekérd 303
"48" - memóriafoglalás 303
"49" - lefoglalt memória felszabadítása 304
"4A" - lefoglalt memóriablokk méretének megváltoztatása 304
"4B" - program betöltése végrehajtatással vagy anélkül 305
"4C" - a program befejezése 306
"4D" - a hívott program befejezési kódjának lekérdezése 307
"4E" - elérési úttal megadott file keresése az első megfelelő file-ig 307
"4F" - további, a megadottnak megfelelő file-ok keresése, 4E hívás után 308
"50-53" - a DOS számára FENNTARTOTT funkciók 308
"54" - VERIFY kapcsoló értékének lekérdezése 308
"55" - a DOS számára FENNTARTOTT funkció 309
"56" - file átnevezése 309
"57" - file dátumának létrehozása és lekérdezése, időpontjának beállítása 310
"58" - memória helyfoglalás módjának lekérdezése illetve beállítása 310
"59" - a hiba leírásának lekérdezése 311
"5A" - file létrehozása egyedi névvel 312
"5B" - új file létrehozása 313
"5C" - file zárolása 314
"5E" - hálózati funkciók 314
"5F" - hálózati hozzárendelések beállítása, lekérdezése 315
"62" - programszegmens lekérdezése 316
"65" - kibővített ország-specifikus információk lekérdezése 317
"66" - globális kódlap beállítása/lekérdezése 318
"67" - periféria-kezelők maximális számának megadása 318
"68" - puffer kiürítése 318

FÜGGELÉKEK

A. A DOS üzenetei 319
A.1 Perifériákkal kapcsolatos hibaüzenetek 319
A.2 A DOS egyéb üzenetei 322
B. Parancsok, alparancsok csoportosított betűrendes listája 397
C. Parancsok, alparancsok formája 400
D. File-név kiterjesztések 404
E. Eszköznevek listája 404
Tárgymutató 405
Irodalomjegyzék 409

13. A PC DOS belső struktúrája

A PC DOS működésbe lépésétől kezdve a következő feladatokat hajtja végre:

- Betölti és működteti a perifériakezelő rendszerét.
- Feldolgozza és végrehajtja az aktuális konzolról érkező parancsokat.
- Kezeli és gazdálkodik a személyi számítógép perifériáival és központi memóriájával (azaz a gép erőforrásaival).
- Igény esetén betölti és elindítja a megfelelő felhasználói programokat.
- A felhasználói programok számára biztosítja a periféria kezeléséhez szükséges rutinrendszert..

Ezen feladatok elvégzését a DOS három, önálló file-ban található fő komponense az IBMBIO.SYS, IBMDOS.SYS és COMMAND.COM végzi. A IBMBIO.SYS és IBMDOS .SYS minden rendszerlemezen kötelezően az első két file-ként helyezkedik el. A rendszer lemezre írásakor ezek rejtett file-ok, ezért sem a DIR, sem a COPY, sem a DEL paranccsal nem kezelhetők.

A három főrészt kiegészíti a rendszerlemez 0. szektorában (a lemez felépítését lásd a "Lemezek felépítése" című fejezetben) található betöltő (boot) program. A DOS részét képezik továbbá a külső parancsok (ASSIGN, ATTRIB, BACKUP, COMP, DEBUG stb.) file-jai. A COMMAND.COM és a külső parancsok megléte a DIR parancs segítségével ellenőrizhető.

13.1 A parancsfeldolgozó program (COMMAND.COM)

A felhasználó a DOS rendszer ezen részével találkozik leggyakrabban. A rendszertöltés és inicializálás után ez a programrész kapja meg a vezérlést.

Feladatai:

1. Kapcsolattartás a felhasználóval:
   - Várakozási állapotnál bejelentkezés az aktuális prompt-tal.
   - Az operátori parancs beolvasása a konzolról, és ennek ellenőrzése.
   - A helyesen megadott parancsok végrehajtása esetén a programok indítása.
2. Perifériák működése közben fellépett hibák kezelése és üzenet küldése a konzolra (24-es megszakítás).
3. A DOS belső parancsainak (CHDIR, DIR, DEL stb.) végrehajtása.
4. <Ctrl-Break> billentyűk hatására keletkező megszakítás kezelése (23-as megszakítás).
5. Programbefejezés után visszatérés a DOS várakozási állapotba (22-es megszakítás).

A program négyféle módon léphet működésbe:

1. Várakozó állapotban a felhasználó a konzolról parancsot ad.

2. A gép bekapcsolása vagy <Ctrl-Alt-Del> egyidejű lenyomása után a DOS iniciáló része betölti a COMMAND.COM-ot és átadja számára a vezérlést.

3. A 2-es, 4-es, 5-ös feladat esetén a DOS törzse hívja meg.

4. A DOS ezen része úgy van szervezve, hogy újólag többször a tárba tölthető és indítható, ezért a felhasználói programok részére a DOS biztosítja a lehetőséget, hogy a 4B függvényhívással betöltsék és elindítsák. Ekkor a felhasználói programnak módjában áll a COMMAND.COM részére előre elkészített parancs átadására. Ilyen módon olyan felhasználói programok írhatók, melyek kezelik a DOS parancsokat is. Így működik például a BASICA-ban a SHELL utasítás.

Az utoljára betöltött parancsfeldolgozó programból az EXIT paranccsal tudunk visszalépni abba a programba, amely meghívta.

Az említett példánál a programok elhelyezkedése a memóriában a következő:

00000	Megszakítási tábla
	DOS állandó részei
	DOS adatterület, pufferek, felhasználói periféria kezelők
	DOS parancsfeldolgozó (COMMAND.COM rezidens része) 22H, 23H, 24H megszakítások kezelése
	DOS által betöltött rezidens programok
	Felhasználói verem .COM file-oknak (256 byte)
	szabad tárterület
	DOS parancsfeldolgozó (COMMAND.COM tranziens része), belső parancsok, parancs-file feldolgozó
A0000	rendszerterület, képernyő-pufferek
F6000	BASIC ROM területe
FE000	BIOS ROM

13.2 A PC DOS indítása

Minden felhasználóban felmerült már a kérdés, mi is történik a gép bekapcsolása, vagy a "meleg újraindítás" (warm reset: <Ctrl-Alt-Del>) és a DOS bejelentkezése közti néhány pillanatban. Nézzük meg most ezt a folyamatot.

Bekapcsolás esetén a gép a ROM BIOS FFFF:0 címén levő utasítástól indul. (Ez általában egy távoli ugrás a ROM BIOS önellenőrző rutinjába.) Ezután a BIOS program ellenőrzi a számítógép hardware elemeit és memóriáját. Ha ezzel végzett, először az A: egységben levő lemez legelső szektorát próbálja olvasni. Ha ez sikeres, a vezérlés rákerül a beolvasott kódra, s - amennyiben ez DOS lemez volt - ettől a pillanattól kezdve működik a PC DOS. Ha az A: meghajtóban nincs lemez, vagy az olvasás sikertelen volt, akkor - winchesteres gépnél - a winchester vezérlő ROM rutinja megpróbál olvasni az első (általában C: jelű) lemez aktív partíciójából. Ha ez sikeres, a vezérlés átadódik a beolva-

sott kódra, s az a fix lemez aktív partíciójáról próbál rendszert indítani. (Ha nincs aktív partíció, hibaüzenetet kapunk.) Ha a gépben nincs winchester lemez, ill. annak első szektora nem olvasható, a vezérlés a ROM BASIC-hez kerül.

Melegindítás (<Ctrl-Alt-Del>) esetén a hardware és memóriateszt elmarad, a ROM BIOS-nak (pl. winchester ROM-nak) a lemez legelső szektorát olvasó rutinja lép működésbe. Amennyiben gépünket programból szeretnénk újraindítani, bekapcsolási indítást a JMP FFFFH:0 , a melegindítást az INT 19H assembler utasítással válthatjuk ki.

Ezután a betöltő program ellenőrzi, hogy a lemezen megvan-e valamennyi rendszer-file. Hiány esetén hibaüzenetet küld a konzolra és leáll. Ha a rendszer-file-ok rendben voltak, akkor betölti az IBMBIO.COM file-t a memória aljára rezidens (állandóan ottmaradó) programként, és átadja neki a vezérlést.

Az IBMBIO.COM file-ban található programok hozzák a DOS-t tényleges indulási állapotba. Ezek a következő műveleteket végzik el:

- Betöltik a lehető legalacsonyabb címre az IBMDOS.COM rendszer-programot.
- Felépítik a DOS futásához szükséges belső táblákat.
- Inicializálják a DOS perifériakezelő rutinjait (handler-eket).
- Megnézik, hogy az aktuális lemezen van-e CONFIG.SYS file. Amennyiben igen, beolvassák, és annak alapján újra konfigurálják a rendszert. (A rendszer konfigurálásával kapcsolatos tudnivalókkal már a megfelelő fejezetben megismerkedhettek az olvasók.)
- Ezután kerül sor a COMMAND.COM betöltésére és elindítására.

A COMMAND.COM elindulása után, ha nem talál AUTOEXEC.BAT nevű file-t és nincs kijelölve beindítható program, akkor kiírja a verziószámot, bekéri a dátumot és időt (lásd a VER, a TIME és a DATE parancsokat). Ezt követőleg az aktuális prompt-tal bejelentkezik.

Ha van indításra kijelölt program, vagy AUTOEXEC.BAT file, akkor nem kerül sor verziószám, dátum, idő kiírásra, hanem a kijelölt programot indítja el a DOS.

Röviden áttekintve a fentieket, az indítás menete:

1. BIOS ellenőrző rutinok lefutása.

2. A BIOS beolvassa és elindítja a DOS betöltő rekordját.

3. A betöltő (boot) program beolvassa az IBMBIO.COM (vagy IO.SYS) file-t.

4. Az IBMBIO.COM betölti a rendszer további részeit, és azokat futáskész állapotba hozza.

5. CONFIG.SYS file létezése esetén az inicializáló program (IBMBIO.COM vagy IO.SYS) értelmezi azt, és annak megfelelően módosítja a rendszert.

6. A parancsfeldolgozó átveszi a vezérlést és az inicializálási ága lefut. (Bejelentkezés vagy kijelölt programok végrehajtása.)

A beindulás után a programok és főbb rendszerterületek tárbeli elhelyezkedését a "PC DOS memóriakezelése, munkaterületei és vezérlőblokkjai" (15. fejezet) című részben található memóriatérkép mutatja.

13.3 A PC DOS-nak a felhasználói programok által elérhető szolgáltatásai, függvényhívási- és megszakítási rendszere

Az előzőekből megismerhettük az operációs rendszer beindulását, a korábbi fejezetek pedig részletesen tárgyalták azt, hogy milyen közvetlen szolgáltatásai, parancsai vannak a PC DOS-nak. Magasszintű programnyelvek, célprogramok használata esetén ezek az ismeretek tökéletesen elégségesek a számítógép kezeléséhez. Igényes programok készítése során vagy software problémák megoldásakor viszont nagy segítség a DOS-nak a felhasználói programok számára nyújtott szolgáltatásainak ismerete. Ezek részletezése előtt bevezetjük a legszükségesebb gépi kódú programozási ismereteket.

13.3.1 A 8086-os processzor regiszterei

Az 1.2 fejezetben szó volt róla, hogy a PC DOS az Intel 8086 típusú és azzal kompatibilis mikroprocesszor alapú gépek operációs rendszere. Az ilyen processzorok a következő regiszterekkel rendelkeznek:

- 4 darab 16 bites általános regiszter: AX, BX, CX, DX. Ezek mindegyike 2 db nyolcbites regiszterként is elérhető. A felső nyolcbites rész jele H (high), az alsóé L (low), azaz például a MOV AX,0201H assembler utasítás egyenértékű a

  MOV AH,02H
  MOV AL,01H

utasításpárral.

- 4 db 16 bites szegmens regiszter: DS, ES, SS, CS. Ezen regiszterek a címképzésben vesznek részt. (DS = Data Segment, ES = Extra Segment, SS = Stack Segment, CS = Code Segment)
- 2 db 16 bites index regiszter: DI, SI. Az SI (Source Index) a string műveletekben a forrásterületre, a DI (Destination Index) pedig ilyenkor a célterületre mutat.
- 2 db 16 bites mutató regiszter: BP, SP. A BP (Base Pointer) regiszter segítségével a tár tetszőleges területén egyaránt futó ún. relatív vagy eltolható programok írhatók. (Relatív címképzés.) Az SP (Stack Pointer) a felhasználói veremtár eltolás értékét tartalmazza.
- Az IP regiszter mindig a következő végrehajtandó utasítás címére mutat.
- Az egy db 16 bites jelző (flag) regiszter felépítése:

Bit	Jelentés
15-12.	fejlesztésre fenntartva
11.	overflow - túlcsordulás
10.	direction - ciklusirány (növekvő vagy csökkenő)
9.	interrupt - megszakítás letiltva/engedélyezve
8.	trap - processzor lépésenkénti utasításvégrehajtás
7.	signum - előjel, ha 0, akkor az eredmény pozitív

6. zéró - eredmény nulla volt-e
5. fejlesztésre fenntartva
4. auxiliary carry - aritmetikai túlcsordulás jelzése
3. fejlesztésre fenntartva
2. parity - eredmény páros volt-e
1. fejlesztésre fenntartva
0. carry - túlcsordulás jelzése

A továbbfejlesztésre fenntartott bitek értéke mindig 0. A többi bit értékei a rájuk hatással lévő utasítások eredményeként változnak. Az, hogy melyik flag-et melyik utasítás állítja, a kézikönyv III. kötetében ismertetjük.

13.3.2 A 8086-os processzor címképzési rendszere

Ez a processzorcsalád valamennyi címet (utasítás-, verem-, adatcím) két részből képzi. A cím szegmense megmondja, hogy a gép memóriájának melyik 64 Kbyte-os területében vagyunk. A cím offset-je (eltolási értéke) mutatja azt, hogy a szegmens által megadott területen belül hányadik byte-ról van szó. A gyakorlatban ez úgy valósul meg, hogy a címzésben résztvevő szegmensregiszter értékének 16-szorosát a processzor hozzáadja a cím relatív helyét jelölő offset értékhez.

Mivel a 16-tal való szorzás a hexadecimális (16-os alapú) számrendszerben a szám végére egy zérus írását jelenti, ezért a címekkel való számolásnál célszerű a hexadecimális számok használata.

A hexadecimális számrendszerben 16 db számjegy van:

0-9-ig a megszokott számjegyek, továbbá
A = 10,
B = 11,
C = 12,
D = 13,
E = 14 és
F = 15

jelentéssel, azaz egy hexadecimális számban a 0-9, A, B, C, D, E, F jelek szerepelhetnek. A felismerhetőség miatt a nem igazi számjeggyel kezdődő ilyen számot szokás 0-val kezdeni.

Például BABA helyett 0BABA írása mutatja, hogy számról van szó. A hexadecimális számrendszer jelzése többféleképpen történhet. Szokásos jelölési formák:

X'BABA', 0BABAH (Macro Assembler),
&HBABA (BASIC),
$BABA (PASCAL).

(Ennek a számnak a decimális értéke:

11*4096 + 10*256 + 11*16 + 10 = 47802)

Mivel egy byte 2^8 = 256 érték megkülönböztetésére ad lehetőséget, a 256 pedig éppen

megegyezik 100H-val, ezért egy byte tartalma mindig megadható egy 2-jegyű hexadecimális számmal. (Pl.: 1111 1111B = 255 = 0FFH.) Hasonló megfontolással látható, hogy két byte tartalmának megadása hexadecimálisan 4-jegyű számok használatát igényli.

Mivel a továbbiakban mindig hexadecimális számokkal fogunk számolni, ezért, ha külön nem jelöljük, akkor mindig ilyet kell érteni, még akkor is, ha nem tartalmazza az A-F számjegyeket.

Első gyakorlatként nézzük meg a címképzést (hexadecimális számokkal). Legyen egy cím szegmens értéke 80, az offset értéke pedig 210. Ezt tömören így jelöljük: 80:210. (Ezt az írásformát használja a DEBUG program is, ezért célszerű ezt megszokni.) A fenti jelölés a

80 * 10 + 210 = 800 + 210 = 0A10

abszolút címet jelöli. Hasonló számolással látható, hogy

1000:0 = 10000 abszolút cím = decimálisan 65536 = 64 Kbyte

azaz a szegmensregiszterek segítségével 16*64 Kbyte, vagyis 1 Mbyte címezhető. Elvileg ez a 8086 alapú PC DOS által maximálisan kezelhető tártartomány.

Nem szóltunk még arról, hogy mikor melyik regiszter tartalmával számolja ki a processzor a címet. Általános szabály, hogy ha másként nem írjuk elő, akkor műveleti címképzésben mindig a CS, adatcímképzésben a DS, a verem címének a számolásában az SS, és sztring műveleti címeknél az ES és/vagy DS regiszter vesz részt.

Ha ettől el akarunk térni, akkor az utasítás elé írt úgynevezett regiszterprefix-szel (CS: vagy DS: vagy ES: vagy SS:) jelezhetjük, melyik regisztert akarjuk használni. Például:

MOV AX,[305]

utasítás betölti az AX regiszterbe a DS:305 és DS:306 memóriacímek tartalmát, de

CS:MOV AX,[305]

a CS:305, CS:306 memóriacímek tartalmát tölti AX-be.

13.3.3 A megszakítás (interrupt) fogalma

A DOS megszakítási rendszerének megértéséhez példákat is fogunk adni. E példákhoz elég lesz a MOV utasítás ismerete, ennek hatása a gépen is jól követhető.

Azok érdekében, akik még nem programoztak 8086 assemblerben, teszünk egy kitérőt, és ismertetjük a MOV-ot, amely már a fenti példában is szerepelt.

A MOV utasítás 8 és 16 bites adatok mozgatására használható, és az utasításkódon kívül két operandust tartalmaz. Az adatmozgatás történhet regiszterek, regiszter és memória, memória és regiszter, valamint két memóriacím között. A második operandus, amely mindig a forrást jelöli, lehet konstans is. Az operandusok a következő megkötésektől eltekintve szabadon alkalmazhatók:

- a két operandusnak egyforma hosszúnak kell lennie (nem mozgathatunk 16 bites regiszterbe 8 bites regisztert),
- nem lehet mindkét operandus szegmensregiszter,
- a cél operandus nem lehet a CS regiszter,
- minden olyan esetben, amikor valamelyik operandus memóriatartalom, és nem egyértelmű, hogy 8 vagy 16 bites műveletről van-e szó, az operandusok előtt a BYTE PTR vagy a WORD PTR prefix-szel jelezni kell a művelet típusát.

Példa:

```
MOV DS,AX              ; 16 bites, DS-be betölti AX tartalmát
MOV AL,CH              ;  8 bites, AL-be betölti CH tartalmát
MOV [30A],DX           ; 16 bites DS:30A,30B címekre tárolja DX tartalmát
MOV [30A],BL           ;  8 bites  DS:30A címre tárolja BL tartalmát
MOV CX,1000            ; 16 bites  CX-be ír 1000-t
MOV BYTE PTR [230],0   ;  8 bites, a DS:230-at kinullázza.
```

Példák hibás MOV utasításokra:

```
MOV  [230],[400]       ; nem megengedett operanduspár
MOV  [230],100         ; nem megállapítható a művelet típusa
MOV  100,AX            ; konstans érték nem lehet az első operandus
MOV  DS,CS             ; szegmensregiszterek között a közvetlen adatmozgatás
                       ; nem megengedett
MOV  AX,CH             ; operandusok hossza eltér
```

Nem foglalkozunk a logikai, aritmetikai és sztringmozgató műveletekkel. A vezérlést irányító utasítások (feltétlen és feltételes ugrások, közeli és távoli szubrutinhívások, software interrupt hívás) közül csak a software interrupt hívás ismerete szükséges a PC DOS funkcióhívási és megszakítási rendszerének megértéséhez. Ezért a továbbiakban ezt ismertetjük.

Először ismerkedjünk meg az interrupt (megszakítás) fogalmával. A számítógép működéséhez mindenképpen szükséges, hogy bizonyos feltételek fellépése esetén félbeszakítson - megszakítson - minden műveletet, és a félbeszakítást kiváltó esemény szerint cselekedjen, majd folytassa az eredeti programot. A programnak azt a helyét, ahol félbeszakadt, megszakítási pontnak hívjuk. Jól érzékelhető a megszakítás a billentyűzet esetében. Tudjuk, hogy egy program futása a <Ctrl-Break> billentyűvel általában félbeszakítható. Ez azért lehetséges, mert minden billentyű leütésekor megszakítás lép fel. Az aktuális program futása felfüggesztődik, működésbe lép a billentyűmegszakítást kezelő rutin, amely kiolvassa a leütött billentyű pozícióját, és a módosító billentyűk helyzete szerint kialakítja a megfelelő kódot, amit a billentyűzet-pufferben tárol. Ha nem érzékelt speciális kezelést igénylő billentyűkombinációt, akkor visszatér a megszakítási pontra, különben a kijelölt feladat szerint cselekszik, például visszatér a DOS-hoz, vagy rendszert tölt, vagy képernyőt nyomtat és csak azután folytatja a programot.

Nézzük most végig lépésekre bontva egy megszakítás lefolyását:

1. Fellép egy megszakítási igény.

2. A processzor megvizsgálja, hogy a megszakítás engedélyezett-e. (Ez programból is befolyásolható, mégpedig a STI = megszakítás engedélyezése és CLI = megszakítás tiltása utasításokkal.)

3. Engedélyezett megszakítás esetén az alábbi módon meghívásra kerül a megszakítás sorszáma szerinti megszakítás rutin. A memória 0:0 címétől kezdődően egy címtáblázat található. A táblázat minden eleme 4 byte hosszú, és egy-egy cím offsetjét és szegmensét tartalmazza. A processzor ezen táblából veszi a meghívandó rutin címét. Mielőtt ennek a rutinnak átadná a vezérlést, a veremtárba elrakja az állapotregisztert, a következő utasítás szegmensét, és a címét. Ez a vezérlésátadási folyamat software úton is előidézhető az INTn utasítással. Az utasításban az n tetszőleges (FF-nél kisebb, vagy egyenlő) szám lehet. A processzor ilyen számú megszakítás-hívást hajt végre. Például az INT 21 hatására a megszakítás vektortábla 21. eleméről (0:84-87) veszi az ugrási címet.

4. A megszakítás-kezelő rutin, miután elvégezte a feladatát, kiadja az IRET utasítást. Ennek hatására a vezérlés visszakerül a veremtárban található szegmens és offset által meghatározott címre úgy, hogy közben mind az állapotregiszter, mind a veremmutató (stack pointer) visszaáll eredeti értékére.

Az interrupt vektortábla elemei közül a 0-0F (a 3-as és az 5-ös kivételével) megszakítások hardware feltételek kezelését végzik. A továbbiak az alsószintű perifériakezelő rendszer, a BIOS egyes rutinjainak a meghívására, illetőleg az ezekhez szükséges paraméterek tárolására szolgálnak.

Az egyes megszakításokat külön fejezetben tárgyaljuk, a DOS funkciókkal együtt.

A fent elmondottakat IBM PC számítógépünkön a következőképpen nézhetjük meg:

```
A>DEBUG                    elindítjuk a debug programot
-d0:0040,44                parancs a megszakítási vektor 10. elemeinek kiírására
000:0040  65 F0 00 F0      (a DEBUG válasza)
-a100                      parancs: cs:100-tól utasítás beírása
1059:100 int 10            (az int 10 utasítást írtuk be)
1059:102 int 3             (az int 3 utasítást írtuk be)
-r                         (regiszterek kiíratása)
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1057 ES=1057 SS=1057 CS=1057 IP=0100 NV UP EI PL NZ NA PO NC
1057:0100 CD10          INT 10
-t                         (az INT 10 utasítás indítása - trace)
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFE8 BP=0000 SI=0000 DI=0000
DS=1057 ES=1057 SS=1057 CS=F000 IP=F065 NV UP DI PL NZ NA PO NC
F000:F065 E99D45        JMP 3605
```

ez a következő utasítás; látható, hogy címe megegyezik a kiíratott vektortábla elem tartalmával

```
-q                         kilépés a DEBUG programból
```

13.3.4 A DOS megszakításainak alkalmazása felhasználói programban

A DOS a saját feladatainak elvégzéséhez a 20-3F megszakításokat használja. Ezek közül

- a program befejezését (20, 22, 27),
- a <Ctrl-Break> kezelését (23),
- a funkcióhívások alatt fellépő hibák kezelését (24),
- a fizikai lemezkezelést (25, 26) és
- a rendszer (funkció) hívást (21) végzők

egyértelműen definiáltak és leírásuk alapján szabadon használhatók. Ez azt jelenti, hogy a felhasználó a leírásban megadott paramétereket beállítja a megfelelő regiszterekben, és kiadja a megfelelő megszakítás hívást. A hívást követő utasításokkal kell a hívás után a regiszterek tartalmát feldolgozni, ellenőrizve a hibátlan lefolyást, illetőleg megállapítani az esetleges hiba okát.

Konkrét példán nézve:

Az A: lemezegységben lévő lemez második logikai szektorának beolvasása a DS:1000 címre a következő assembler utasításokkal lehetséges:

```
MOV AL,0        ; A egység beállítása
MOV CX,1        ; 1 szektor olvasása
MOV DX,1        ; első szektor
MOV BX,100      ; pufferterület első címe
INT 25          ; olvasás lemezről (fizikai)
.
.
.
```

Az INT 25 befejeződése után a ds:1000 címen meg fog jelenni az A egységben lévő lemez 0. oldalán lévő második szektor tartalma. (Az abszolut szektor sorszámozás zéróval indul - lásd a lemezek felépítésével foglalkozó fejezetet.)

Az egyes megszakítások leírása és részletes ismertetése a DOS általános struktúráját leíró fejezetek után, a 16. fejezetben található.

Végezetül felhívjuk a figyelmet arra, hogy a 22 és a 23 megszakítások kezelése eltér, ugyanis itt a felhasználónak kell igény esetén a megfelelő címeket az interrupt vektortáblában elhelyeznie. Az így elhelyezett címekre a megfelelő feltétel teljesülésekor a DOS adja át a vezérlést.

Hasonlóan nem hívható a felhasználó által a 24-es interrupt sem. Ez úgy használható, hogy a felhasználó a leírások alapján elkészíti a saját hibakezelő rutinját, és ennek a rutinnak a címét helyezi el az interrupt vektor megfelelő elemében. Így hiba esetén a DOS a felhasználó rutinját fogja meghívni. (Az interrupt vektor egyes elemeinek kicserélése a 25 és 35 alfunkcióhívásokkal oldható meg a legegyszerűbben.)

13.3.5 A funkcióhívási rendszer

Az előző alfejezetből látható, hogy a DOS perifériák kezeléséből és a felhasználóval, a felhasználói programmal való kapcsolattartásból csak néhány igen alapvető dolgot old meg önálló megszakítással. A legtöbb feladatot az egyetlen kiemelt megszakítás (a 21-es) megfelelő paraméterezésével végezhetjük el. Ez a megszakítás a DOS funkcióhívás, másnéven a rendszerhívás. Ez igen sokrétű szolgáltatást biztosít, és sok feladat (pl. kiírás képernyőre, file-kezelés, programbefejezés) többféle módon is megoldható. Ennek elsősorban az az oka, hogy az újabb javított, bővített verziók a kompatibilitás miatt változatlanul megtartják a korábbi hívásokat, ugyanakkor a bővítés sokszor csak a korábbi feladatok újabb módon való megvalósításával volt lehetséges.

Felmerült például az igény, hogy a program lefutása után meg lehessen állapítani a sikeres vagy sikertelen működést. Erre vezették be a 2.0 verziótól kezdődően a 4C eljárást, a program befejezése funkcióhívást. Ez használható a 0 funkcióhívás helyett, de több lehetőséget nyújt annál.

A funkcióhívások közös jellemzője, hogy a hívás sorszámát mindig az AH regiszternek kell tartalmaznia. A többi paramétert a megfelelő regiszterekben kell megadni. Az AH és a többi regiszterek beállítása után az INT 21 kiadásával lehet a megfelelő funkcióhívást aktivizálni.

Például nézzük meg a DOS verziószám lekérdezését, azaz azt, hogy a program milyen verziójú DOS alatt fut. Erre a 30-as funkcióhívás szolgál, amelynek nincs bemenő paramétere. Az aktuális verziószámot az AX-ben kapjuk vissza. A szükséges utasítások:

```
MOV AH,30
INT 21
```

a visszatéréskor az AX-ben megkapjuk a verziószámot.

13.4 A DOS perifériakezelése

Az előző alfejezetben bemutattuk, hogy milyen módon szervezi az operációs rendszer a munkáját, és hogy ebbe hogyan kapcsolódhat be a felhasználó. Most rátérünk a perifériák (eszközök, berendezések) kezelésére. A DOS kétféle módon, karakteresen és blokkosan kezel perifériát.

13.4.1 Karakteres berendezések

Ezeknél az eszközöknél a DOS az írás-olvasási műveleteket karakterenként végzi. Az egyes berendezések megkülönböztetése egy maximum 8 karakter hosszú névvel történik. A rendszer karakteresen kezelt perifériáinak nevei:

CON - konzol I/O egység
NUL - „üres" periféria
AUX - soros vonali I/O egység

COM*i* - soros vonali I/O egység. *i* = 1, 2, 3 vagy 4. A 3 illetve 4 érték csak a DOS 3.30 verziótól kezdve használható.
CLOCK$ - rendszeróra
LPT*i* - párhuzamos vonali kimenet. *i* = 1, 2 vagy 3.

Az első kettő kivételével a karakteres berendezések használhatósága a gép kiépítettségén is múlik.

Ezeken kívül a felhasználó maga is definiálhat ilyen perifériákat. Ekkor egyrészt el kell készítenie a megfelelő perifériakezelő rutinrendszert megfelelő formában, másrészt a készülék nevét, a kezelőrutint tartalmazó file nevét és a rutin belépési pontjait közölnie kell a DOS-szal. (A CONFIG.SYS file DEVICE=... parancsában, illetőleg a perifériakezelő rutint tartalmazó file elején a DOS által megkövetelt formában.)

Mivel saját perifériakezelő rutin definiálásának bonyolult szabályai vannak, ezért azokkal külön rész foglalkozik.

13.4.2 Blokkos berendezések

A rendszer a lemezegységeket (hajlékony és fix lemezegységek) blokkos átvitellel kezeli, azaz tényleges fizikai adatátvitelt csak meghatározott (pl. 1 Kbyte) adatmennyiséggel hajt végre. A felhasználó írási/olvasási igényeit a DOS pufferezés segítségével elégíti ki. Ez azt jelenti, hogy ha egy lemez-file-ba karakterenként visszük is ki az adatokat, akkor az ténylegesen csak az 1024. byte kiírása után kerül fel a lemezre. Az utolsó töredék blokk kiírása mindig a file lezárásakor történik.

A blokkos perifériákat a DOS sorszámmal azonosítja, azonban a könnyebb kezelhetőség miatt az operátori parancsokban (DIR, DEL, REN stb.) a sorszámokat az ABC megfelelő betűje (A = 0, B = 1, ...) helyettesíti.

Saját perifériakezelőt blokkos berendezésekre is lehet készíteni. Ezekre is érvényesek a karakteres berendezéseknél fent felsorolt szabályok.

14. Lemezkezelés és a felhasználói perifériakezelők szerkezete

Ez a fejezet a PC DOS operációs rendszer perifériakezelését ismerteti.

Első részében a leggyakoribb külső adattárolók, a cserélhető lemez-(floppy) egység és a fix, vagy másként winchester lemezegység (Winchester disk/fix disk drive) logikai felépítését, táblázatait találhatja az Olvasó.

A fejezet második része a felhasználó által készíthető perifériakezelő rutinokkal kapcsolatos ismereteket foglalja össze.

14.1 A hajlékony és a fix lemezek felépítése

A DOS operációs rendszer igen nagy előnye, hogy az MS DOS és PC DOS rendszer alatt előállított lemezek szinte géptípustól függetlenül olvashatók, írhatók. Ez annak köszönhető, hogy a rendszer szerves részét képezi a lemezek egységes formátuma.

Mielőtt ezt ismertetnénk, először nézzük meg a lemez fizikai felépítését:

A lemezen az információ körkörösen, úgynevezett sávokon (track) helyezkedik el. A koncentrikus köröket alkotó sávokon belül floppy-nál lemezformátumtól függően 8 vagy 9 darab 512 byte-os szektor található (winchesternél több szektor helyezkedik el a sávokon). Ez a lemezen az információtárolás fizikai alapegysége.

A lemez lehet egy- vagy kétoldalas; szimpla, dupla, illetve négyszeres sűrűségű és nagy kapacitású.

Az egyoldalas lemeznek csak az alsó felén van információ. A szimpla sűrűségű lemezen 20, a duplán 40, a négyszeres sűrűségűn és a nagy kapacitásún 80 sáv található. (Megjegyzés: a forgalomban lévő gépek nem használnak szimpla sűrűségű lemezt.)

A winchester lemezben több, a cserélhető lemezhez hasonló, egymáshoz erősített mágneses lemez található. Ezeken mind a lemezoldalak, mind az egy oldalon található sávok, mind pedig a sávokon belüli szektorok száma több lehet, mint a cserélhető lemez megfelelő értékei.

Ahhoz, hogy a lemezen egy adatot megkeressünk, három jellemzőt kell megadni:

- az adatot tartalmazó sávot,
- az oldalt és
- a szektort.

A DOS az egységes kezelést úgy valósítja meg, hogy eltekint a szektorok tényleges elhelyezkedésétől, úgy veszi, mintha ezek folytonosan helyezkednének el. Így képez egy 0-tól induló logikai sorszámot az adatterületek (szektorok) azonosítására.

A logikai sorszámot a következőképpen kell elképzelni:

A 0. oldalon található 0. sáv első szektora az 1-es, második szektora a kettes, majd ez így

folytatódik a sáv utolsó szektoráig. A következő sorszámot egyoldalas lemez esetén az 1-es sáv 1-es szektora, míg kétoldalas lemez esetén az 1-es oldal 0. sáv 1-es szektor kapja. Kettőnél több oldalas (több fejes) Winchester lemezeknél az 1-es oldal 0-ás sávja következik. Ilyen rendszer szerint folytatódik a szektorok sorszámozása. Az utolsó sorszám az 1-es - illetőleg winchesternél a legmagasabb sorszámú - oldal utolsó sávjának az utolsó szektorát jelzi.

Ezek után tekintsük át a hajlékony (floppy) lemezek felépítését:

0. szektor: **DOS boot szektor.** Az itt elhelyezkedő töltőprogram arra szolgál, hogy rendszertöltés esetén betöltse a lemezről a DOS további részeit és elindítsa azokat,
1. szektor: Innen kezdve k szektoron keresztül található az ún. **file leíró tábla** (FAT = File Allocation Table)
1 + k. szektor: Innen kezdve a file leíró tábla másolata található.
1 + 2k. szektor: Ettől kezdődően helyezkedik el a főtartalomjegyzék.
1 + 2k + n. szektor: Ettől kezdve találhatók a DOS és a felhasználó file-jai.

A winchester-ek felépítése kicsit összetettebb, ugyanis nagyobb kapacitása miatt biztosították a lemez (legfeljebb négy) részre oszthatóságát. Egy-egy összetartozó rész neve partíció.

A partíciók számát és méretét a felhasználó állíthatja be az FDISK parancs segítségével. (Ennek részletes leírása az 5.1 fejezetben található.)

A több partícióra való oszthatóság lehetősége elsősorban azt a célt szolgálja, hogy a winchester-rel ellátott gépet többféle operációs rendszerrel is lehessen használni. Ekkor az egyes operációs rendszerek egy-egy partíciót foglalnak el, mégpedig fizikailag összefüggő területen.

A partíciók kezelése egy úgynevezett partíciós tábla segítségével történik. A partíciós tábla a winchester fizikailag legelső szektorán helyezkedik el, és tartalmazza a partíciók azonosításának és elérésének az adatait.

A DOS partíció szerkezete megegyezik a cserélhető lemezek szerkezetével, azaz a tényleges kezdetétől függetlenül (ez csak a partíciós táblából állapítható meg) a 0. logikai DOS szektorral kezdődik, amely a DOS boot szektort tartalmazza. Ezt követi a file allokációs tábla és ennek másolata.

A rendszerterületek közül az utolsó a főtartalomjegyzék. Az egyes területek kezdete, mérete és típusa a boot szektorban található.

A boot szektor és a partíciós tábla felépítését a következő szakasz ismerteti részletesen.

14.1.1 A boot szektor és a partíciós tábla

A boot szektor, amely a cserélhető lemezen a 0. oldal 0. sáv első szektora, illetőleg a fix lemez esetén a DOS partíció 0. logikai szektora tartalmazza a PC-DOS rendszer betöltését elindító programot. Tartalmazza továbbá a program paramétereiként, meghatározott helyen, mindazokat az adatokat, amelyek a lemezen található DOS táblázatok feldolgozásához szükségesek. Ennek felépítése a következő:

Adatmező helye byte-ban **Adatmező tartalma**

00-02. Egy ugró utasítás a BOOT program további részére. Ez vagy 3 byte hosszú közeli (near) ugrás (E9H) vagy 2 byte-os relatív (short) ugrás (EBH), melyet egy NOP követ.

03-0B. Egy 8 byte hosszú azonosító, mely a DOS nevéből és verziószámából áll.

0B-0C. * Szó, amely egy szektor méretét tartalmazza byte-okban.

0D. * Az egy logikai blokkban lévő szektorok számát tartalmazó byte. Megmutatja, hogy a FAT egy egysége hány szektort foglal magába. Ha tartalma például 4, akkor a lemezen lefoglalt legkisebb terület 512 byte-os szektorméret mellett 4*512 = 2048 byte = 2 Kbyte. (A DOS a tartalomjegyzékben látható file-méretet 2 Kbyte többszörösére fogja kerekíteni és ekkora területet használ el az adathordozóról.)

0E-0F. * Szó, amely a foglalt szektorok számát tartalmazza. Ezek a szektorok a 0., a boot rekord kezdetétől helyezkednek el, és megelőzik a FAT első kópiáját.

10. * A lemezen található FAT másolatok darabszámát (ez általában 2) tartalmazó byte.

11-12. * Szó, amely megadja, hogy a főtartalomjegyzékben maximum hány bejegyzés lehet.

13-14. * Szó, amely az adathordozón elhelyezkedő szektorok darabszámát tartalmazza. (Az adathordozó valamennyi szektora, beleértve a boot szektort, a FAT-ot és a főtartalomjegyzéket tartalmazó szektorokat stb.)

15. * Az adathordozó típusát jelző byte. Szerkezetét és az érvényes típusjelzők listáját a hozzá tartozó formátumokkal együtt később ismertetjük.

16-17. * A FAT egy másolatának mérete szektorokban.

18-19. Szó, amely az egy sávon található szektorok számát tartalmazza.

1A-1B. Szó, amely a lemezoldalak számát tartalmazza.

1C-1F. Dupla szó, amely a rejtett szektorok számát tartalmazza. Ezek a szektorok a DOS boot rekord előtt helyezkednek el, azaz cserélhető lemezen itt nulla áll, winchester-nél pedig a DOS partíciót megelőző szektorok száma.

20-23. Dupla szó. Jelenleg a DOS nem használja. Célszerű felhasználása: ha a 13-14. szó 0, akkor ez a duplaszó jelzi a logikai szektorok számát. Ez 32 Mbyte-os.

24. Az indító (boot) program, melynek tényleges kezdete az első két vagy három byte-on található ugró utasításból állapítható meg.

A csillaggal megjelölt részek együtt alkotják az ún. BIOS paraméter blokkot (BPB). (A kiterjesztett DOS partíció boot szektorának leírása a 14.1.4 részben található.)

A partíciós tábla a Winchester lemez 0. sávjának 0. szektorában található. A szektor elején egy boot program helyezkedik el. Ez akkor kapja meg a vezérlést a ROM BIOS-tól, ha az A: lemezegység üres. A program feladata a rendszertöltésre kijelölt partíció kiválasztása és a kiválasztott partíció boot rekordjának aktivizálása. A partíció boot rekordja a továbbiakban ugyanúgy működik, mint a cserélhető lemezeken található boot program. (A rendszertöltés menete megtalálható "A PC-DOS belső struktúrája" című 13. fejezetben.) A boot rekordot követően az **1BE címtől kezdődően található** a partíciós tábla, amelynek felépítése a következő:

Mező címe byte-ban	**Mező tartalma**
1BE	Az első rendszer töltést jelző byte-ja. Ennek tartalma alapján keresi ki a boot rekord a betölthető partíciót. Lehetséges értékei: • betölthető partíció esetén - hexadecimális 80, • nem boot-olható partíciónál - 00 (A négy közül csak egy partíció tartalmazhat 80-at.)
1BF-1C1	Az első partíció kezdetének címe a lemezen. Első byte tartalma: lemezoldal Második byte tartalma: szektor Harmadik byte tartalma: sáv
1C2	Rendszer jelző byte: • 00 ismeretlen • 01 elsődleges DOS 12 bites FAT • 04 elsődleges DOS 16 bites FAT • 05 kibővített DOS
1C3-1C5	Az első partíció végének címe. Első byte tartalma: lemezoldal Második byte tartalma: szektor Harmadik byte tartalma: sáv
1C6-1C9	Duplaszó, mely a partíció kezdetének logikai szektorcímét tartalmazza.
1CA-1CD	Duplaszó, mely a partíció méretét tartalmazza.
1CE	Második partíció rendszertöltést jelző byte-ja (tartalmát lásd az 1. partíciónál).
1CF-1D1	A második partíció kezdetének címe a lemezen.
1D2	A második partíció rendszer jelző byte.
1D3-1D5	A második partíció végének címe.
1D6-1D9	Duplaszó, mely a második partíció kezdetének logikai szektorcímét tartalmazza.
1DA-1DE	Duplaszó, a második partíció méretét tartalmazza.

1DE	A harmadik partíció rendszertöltést jelző byte-ja.
1DF-1E1	A harmadik partíció kezdetének címe a lemezen.
1E2	A harmadik partíció rendszer jelző byte.
1E3-1E5	A harmadik partíció végének címe.
1E6-1E9	Duplaszó, mely a harmadik partíció kezdetének logikai szektorcímét tartalmazza.
1EA-1ED	Duplaszó, mely a harmadik partíció méretét tartalmazza.
1EE	A negyedik partíció rendszertöltést jelző byte-ja.
1EF-1F1	A negyedik partíció kezdetének címe a lemezen.
1F2	A negyedik partíció rendszer jelző byte.
1F3-1F5	A negyedik partíció végének címe.
1F6-1F9	Duplaszó, mely a negyedik partíció kezdetének logikai szektorcímét tartalmazza.
1FA-1FD	Duplaszó, mely a negyedik partíció méretét tartalmazza.
1FE-1FF	Két byte-os jelző, melynek értéke: 55 AA. Azt jelzi, hogy a szektor a partíciós táblát tartalmazza.

14.1.2 A file leíró tábla (FAT)

A DOS a lemezek foglaltságát az úgynevezett file leíró tábla (File Allocation Table, röviden FAT) segítségével tartja nyilván. Ez a tábla tartalmazza az egy file-hoz tartozó területek láncát is.

A DOS az adatokat a lemezre logikai blokkonként (a DOS szakirodalomban: cluster) viszi ki, és így is tartja nyilván. Egy logikai blokk meghatározott számú (cserélhető lemeznél 1 vagy 2, Winchester-nél 4 vagy 8) szektorból áll.

A FAT 12, illetve 4085-nél több logikai blokkot tartalmazó lemezeknél 16 bites bejegyzéseket tartalmaz. E bejegyzések és a logikai blokkok között egy-egyértelmű megfeleltetés van. Az egyes bejegyzésekhez a velük egyező sorszámú logikai blokk tartozik. Az operációs rendszer a lemezre íráskor megkeresi a FAT-ban az első nem foglalt bejegyzést és az annak megfelelő blokk területére fog írni. Egyidejűleg a FAT-ban bejegyzi a kiírás folytán előállott változásokat.

A FAT első két bejegyzése nincs blokkhoz rendelve. Ezeken a területeken ugyanis a boot rekord, majd a FAT két másolata és a tartalomjegyzék helyezkedik el. A FAT első byteján a lemez típusát jelző byte található. A típusleíró byte felépítése bitekre lebontva:

7-3. bit értéke kötelezően 1.
2. bit = 0 az adathordozó nem cserélhető
= 1 az adathordozó cserélhető

1. bit = 0 az adathordozó nem 8 szektoros
= 1 az adathordozó 8 szektoros
0. bit = 0 egyoldalas lemez
= 1 kétoldalas lemez.

A fenti értelmezés nem hagyott azonosító bit-et a lemezek minden jellemzőjére, pl. méret, felírási sűrűség stb. A PC-DOS 3.30-as verziója a következő standard lemezek használatát biztosítja:

lemez mérete(")	5-1/4	5-1/4	5-1/4	5-1/4	8	8	8
sáv/oldal	40	40	40	40	77	77	77
byte/szektor	512	512	512	512	128	128	128
szektor/blokk	2	2	1	1	4	4	1
fenntartott szektorok	1	1	1	1	1	4	1
FAT-ok száma	2	2	2	2	2	2	2
bejegyzések sz. a főszótárban	64	64	112	112	68	68	192
szektorok száma	320	360	640	720	2002	2002	616
típusleíró byte	FE	FC	FF	FD	FE	FD	FE
szektor/FAT	1	2	1	2	6	6	2
szektor/sáv	8	9	8	9	26	26	8
fejek száma	1	1	2	2	1	1	2
rejtett szektorok	0	0	0	0	0	0	0

lemez mérete(")	5-1/4 3-1/2	5-1/4 3-1/2	5-1/4 3-1/2	5-1/4 3-1/2	5-1/4
sáv/oldal	80	80	80	80	80
byte/szektor	512	512	512	512	512
szektor/blokk	2	2	2	2	4
fenntartott szektorok	1	1	1	1	1
FAT-ok száma	2	2	2	2	2
bejegyzések száma a főszótárban	112	112	112	112	224
szektorok száma	640	1280	720	1440	2400
típusleíró byte	FA	FB	F8	F9	F9
szektor/FAT	1	2	2	3	7
szektor/sáv	8	8	9	9	15
fejek száma	1	2	1	2	2
rejtett szektorok	0	0	0	0	0

A többi bejegyzés tartalma szerint, a következő információkat hordozhatja:

(F)FF8-(F)FFF	A file utolsó blokkját jelzi. (Általában csak az (F)FFF érték használatos.)
(F)FF0-(F)FF7	A hozzá tartozó blokk foglalt, vagy hibás. A fizikai hibás blokkot az (F)FF7 érték jelzi.
(0)000	A bejegyzéshez tartozó blokk szabad, felhasználható.

(n)nnn Ahol (n)nnn egy 1-nél nagyobb, a blokkok maximális számánál kisebb érték. Azt mutatja, hogy a file folytatása hányadik blokkban található. (A blokk egy file részére foglalt.)

A file első blokkjának sorszáma a file-nak megfelelő tartalomjegyzék bejegyzésben található. A fentieknek megfelelően pedig a FAT e sorszámmal egyező sorszámú bejegyzése tartalmazza a file következő blokkjának sorszámát. A megfelelő FAT bejegyzés megint mutatja a folytatást, míg az utolsó logikai blokkhoz tartozó bejegyzésben (F)FFF érték található.

Megjegyzések:

1. A 12 bites bejegyzések folytonosan helyezkednek el, mégpedig úgy, hogy minden 3. byte alsó négy bitje az előtte lévő byte-tal alkot egy számot, és ennek a számnak a magas helyiértékű részét képezi, a felső négy bitje pedig az utána következő byte-tal alkot egy számot, és ennek a számnak az alacsony helyiértékű részét képezi. Például, ha a FAT első hat byte-ján a következő értékek találhatók:

1.	2.	3.	4.	5.	6.
FD	FF	FF	03	F0	FF

akkor ezek a következő bejegyzéseket jelentik:

1. bejegyzés = F FD
2. bejegyzés = FF F
3. bejegyzés = 0 03
4. bejegyzés = FF F

2. A FAT mindig a boot rekord után helyezkedik el, általában két példányban. A mérete, a másolatok száma, és a blokkméret a boot rekord-ban megtalálható (lásd előző szakasz).

3. Fix lemezeknél a partíció táblában van jelezve, hogy egy bejegyzés 12 vagy 16 bitből áll-e.

4. A 12 bites bejegyzések esetén úgy kaphatjuk meg egy blokk helyét a FAT-ban, hogy a blokk sorszámát (a sorszámozás 0-tól indul) beszorozzuk 1.5-tel. Ha az eredmény egész szám, akkor az így kapott sorszámú byte-on, és a rákövetkező byte alsó négy bitjén található a bejegyzés. Tört eredmény esetén a felkerekített érték a teljes, a lefelé kerekített érték pedig a fél byte helyére mutat.

5. 16 bites bejegyzések esetén a logikai rekord sorszámát 2-vel kell beszorozni, és akkor megkapjuk a rekordhoz tartozó bejegyzés kezdő byte-ját.

6. Ha a rekord logikai sorszámából meg akarjuk kapni a rekord relatív szektorcímét (például a 26-os vagy 25-ös DOS interrupt híváshoz, vagy a DEBUG használatához), akkor a következőképpen járjunk el:

   - vonjunk le a blokkszámból 2-t,
   - szorozzuk meg az eredményt az egy blokkban található szektorok számával,
   - adjuk hozzá az adatterület kezdetének relatív szektorcímét.

14.1.3 A tartalomjegyzék (directory) felépítése, altartalomjegyzékek

A lemezeken a file leíró táblát követően helyezkedik el a főtartalomjegyzék, melyet a FORMAT parancs mindig kialakít. A tartalomjegyzékben a bejegyzések száma korlátozott, a maximális értéket a boot rekord tartalmazza.

Az altartalomjegyzékek a lemezen speciális file-ként helyezkednek el.

Mind a főtartalomjegyzékben, mind az altartalomjegyzékben azonos formátumú bejegyzések tartják nyilván a file-okat.

A tartalomjegyzékekben egy bejegyzés 32 (hexadecimális 20) byte-ból áll és a következő a felépítése:

Hely byte	**Tartalom**
0-7	File-név. Ha a 0-dik byte az itt következő értékek valamelyikét veszi fel, akkor a byte a bejegyzés státuszát mutatja és nem file-nevet jelent. A státusz értékek a következők lehetnek: 00 = Olyan bejegyzés, melyet a rendszer eddig még sohasem használt fel. (Az ilyen bejegyzés nyilvántartása a könyvtárban való keresést gyorsítja meg.) E5 = A bejegyzést a rendszer már felhasználta, de azt a file-t, amelyet azonosított, már törölte. 05 = Jelzi, hogy a file-név első karaktere E5H. 2E = A bejegyzés nem file-t, hanem alkönyvtárat azonosít. Ha a következő byte értéke szintén 2E, akkor ez a könyvtár nem a gyökér, hanem van "őse". (Ekkor a blokkszám mező, amelynek leírása itt majd következik, a könyvtár ősének kezdő blokkszámát adja, vagy értéke 0000, ekkor pedig az ős a gyökér könyvtár.) Ha a 0-dik byte értéke nem a fentiek valamelyike, akkor az egyszerűen a file-név első karaktere. (Ha a file-név nem tölti ki a rendelkezésre álló 8 karakternyi helyet, a rendszer balra igazítja.)
8-0A	A file-kiterjesztés.
0B	A file attributum jelző byte-ja. Egyrészt speciális file-ok megkülönböztetésére, másrészt a file-műveletek sikeres befejezésének ellenőrzésére szolgál. A byte egyes bitjeinek jelentése: **bit** **jelentés** 5. Az úgynevezett archív bit. Ha értéke 1, akkor a file az utolsó írásra való megnyitást követően le is lett zárva. A BACKUP és a RESTORE használja fel ezt a bitet, ebből tudják meg, hogy az utolsó mentés óta változott-e a file. Ezt a bitet a rendszer az attributum byte bármely más bitjével együtt is tudja

használni, azaz ettől függetlenül bármely másik bit is használható. Ha csak ez a bit van beállítva, a byte értéke hexadecimális 20.

4. Altartalomjegyzék azonosító bit. Ha értéke 1, akkor a bejegyzés egy altartalomjegyzéket azonosít, így a normál file műveletekben (REN, DEL, COPY stb.) nem vesz részt.

Ez a bit csak a rejtettséget jelzővel kombinálható, így ha az attributum byte alkönyvtárat jelez, értéke hexadecimális 10 vagy 12.

3. Kötet (volume) azonosító jelzése. Ha a bit értéke 1, akkor a bejegyzéshez nem file tartozik, hanem az első 11 byte-jával egy egész lemezt azonosít. Ilyen bejegyzés a /V opcióval indított formattálással, illetve a LABEL paranccsal hozható létre.

Ez a bit nem kombinálható a többivel, így beállítása esetén az attributum byte értéke 8.

Csak a főtartalomjegyzékben fordulhat elő. Ha több ilyen attributumú bejegyzés található, akkor a rendszer az elsőt tekinti kötet-azonosítónak.

2. Rendszer-file jelzése. Ha a bit értéke 1, akkor a file rendszer-file. Ez azt jelenti, hogy a normál DIR parancs nem kezeli, file-műveletekben nem vesz részt.

1. Rejtett (hidden) file jelzése. Ha a bit értéke 1, akkor a file rejtett file. A DIR műveletekben nem vesz részt, de csoportnévvel DOS parancsokból is elérhető.

0. Csak olvasható (read-only) file jelzése. Ha a bit ertéke 1, akkor a bejegyzéshez tartozó file nem írható/módosítható.

Megjegyzések:

1. Ha egyszerre több bit is be van kapcsolva, akkor az egyes bitek magyarázatánál megadott byte értékeket össze kell adni. Például egy lezárt, csak olvasható, rejtett, rendszer-file attributum byte-jának értéke 20 + 4 + 2 + 1 = 27 hexadecimálisan.

2. A 43-as funkcióhívással a következő attributumok megváltoztathatók:

   - csak olvasható file,
   - rejtett file,
   - archív bit.

12-21 A PC-DOS számára fenntartott byte-ok.

22-23 A file létrehozásának, illetőleg utolsó aktualizálásának (update) időpontját tartalmazza, a következő felépítésben:

	23. byte	22. byte
bitek	F E D C B A 9 8	7 6 5 4 3 2 1 0
tartalom	o o o o o p p p	p p p m m m m m

ahol

- óóóóó egy 0-23 közötti bináris szám, amely jelzi az órákat
- pppppp egy 0-59 közötti bináris szám, amely jelzi a perceket
- mmmmm egy 0-30 közötti bináris szám, amely jelzi a másodperceket, kettesével növelve.

24-25 A file létrehozásának, illetőleg utolsó aktualizálásának (update) dátumát tartalmazza a következő felépítésben:

	25. byte							24. byte							
bitek	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
tartalom	e	e	e	e	e	e	e	h	h	h	h	n	n	n	n	n

ahol

- éééééé egy 0-119 közötti bináris szám, amely az évet jelenti (1980-2099)
- hhhh egy 1-12 közötti bináris szám, amely a hónapot jelzi
- nnnnn egy 1-31 közötti bináris szám, amely a napot mutatja.

26-27 A file kezdőblokkjának mutatója, azaz a file első logikai blokkjának blokkszáma. Szó formában van tárolva a file első logikai blokkjának (cluster) címe.

Az adatok számára rendelkezésre álló terület mind a fix, mint a cserélhető lemezeken a 2. blokktól kezdődik.

28-31 A file mérete byte-okban. Duplaszó formátumban előjel nélküli bináris számként tárolódik a file mérete.

14.1.4 Kiterjesztett DOS partíció felépítése

A 3.30-as DOS verzió egyik új eleme a kiterjesztett DOS partíciók használata. Ez lehetővé teszi, hogy a fix lemez(ek)en az ún. elsődleges DOS partíció mellett egy másik, kiterjesztett DOS partíciót is létrehozzunk. Mi több, ennek a nagysága meghaladhatja a 32 Mbyte-ot is.

A kiterjesztett DOS partíciót a rendszer **kötetek**re osztja, s egy-egy kötethez logikai egységet rendel. Fizikailag a DOS a köteteket, mint fix lemezek DOS partícióit kezeli, azzal a különbséggel, hogy ezekből nem tud rendszert tölteni, bár a partíciós táblában tölthető bejegyzés szerepel. A kiterjesztett DOS partícióban levő boot-rekord (az ún. **kiterjesztett boot rekord**) szerkezete megfelel egy 'igazi' boot rekordnak, az alábbi eltérésekkel:

A **rendszer-jelző byte** értéke az alábbiakat jelenti:

- 00 Ehhez a bejegyzéshez nem tartozik lemezterület
- 01 12 bites DOS partíció
- 04 16 bites DOS partíció
- 05 Ez a bejegyzés a következő kiterjesztett boot rekord helyét jelzi
- 06 Fenntartott

A 01,04 és 06 rendszer-jelző byte-ok esetén a partíció kezdetét és végét jelző három-három byte a kötet megfelelő adatait tartalmazza és a kiterjesztett DOS partíció elejétől számítandó. Ha a 05-ös rendszer-jelző byte-ot használjuk, akkor a három-három byte a következő kötet adatait tartalmazza.

A logikai rekordszám a kötet elejétől mérendő, míg a partícióban levő szektorok száma a logikai egységhez tartozó szektorok számát jelenti.

Ettől eltérően, ha a rendszer-jelző byte 05, akkor a logikai szektorszám az egész kiterjesztett DOS partíció elejétől értendő, és a következő kiterjesztett boot rekord helyét mutatja. A szektorszámot tartalmazó duplaszónak ilyenkor nincs jelentése és 0-val kell feltölteni.

A kiterjesztett boot rekord utolsó két byte-ja az 55AAH számot tartalmazza, jelezve, hogy egy 'igazi' boot rekordról van szó.

14.2 A felhasználói perifériakezelő program (handler, driver)

A DOS perifériakezelését ismertető fejezetben már említettük, hogy a DOS lehetőséget biztosít a felhasználónak, hogy saját perifériakezelő rutint használhasson. A megfelelően elkészített felhasználói perifériakezelő rutin beilleszthető az operációs rendszerbe, azaz azt a gépet használók a beillesztés után úgy érzékelik, mint a DOS szerves részét.

14.2.1 A perifériakezelő program felépítése

A perifériakezelő (vagy driver) speciális memóriaképű program. COM, EXE vagy SYS kiterjesztésű file tartalmazhatja. Egy file-ban több, egymástól független perifériakezelő is elhelyezhető.

A file-beli egyes periféria-kezelő programok elején egy-egy táblázat helyezkedik el - az úgynevezett **perifériakezelő fejléc (device header)** -, amely tartalmazza a programhoz tartozó belépési címeket, a kezelendő berendezés típusát, nevét, a következő drivert megelőző fejléc kezdőcímét. (A fejlécet részletezzük majd.) A fejlécet követi a tényleges perifériakezelő program.

A perifériakezelőt leíró programnak nem kell a PSP részére maga előtt hexadecimális 100 byte-ot fenntartania, azaz assemblerben való megírása esetén nem szabad az ORG 100H direktívával kezdődnie a programnak. A perifériakezelőt ugyanis a DOS egyszerű file-ként, PSP nélkül tölti be.

A DOS a felhasználói perifériakezelő létezéséről a CONFIG.SYS file-ból szerez tudomást, ugyanis, ha saját perifériakezelőt akarunk alkalmazni, a CONFIG.SYS file-nak tartalmaznia kell egy **DEVICE=***file-név* paramétert. A file-név a felhasználói perifériakezelőt tartalmazó file neve. Természetesen a DOS csak új rendszertöltés után fogja a perifériakezelőt felismerni, nem pedig a CONFIG.SYS átírása után. (A témához kapcsolódó információk találhatók a "Rendszer konfigurálása" és a "PC-DOS belső struktúrája" című, 11. és 13. fejezetekben.)

A perifériakezelő fejlécének előírt szerkezete a következő:

Cím byte	Adattartalom
0-3	Duplaszó, amely a file-on belül rámutat a következő perifériakezelő program fejlécére. Ha ez a program a file-ban az utolsó, akkor az első két byte értéke FFFF. Ha több program van egy file-ban, de azok egy szegmensen belül helyezkednek el, akkor a második két byte értéke FFFF.
4-5	Az úgynevezett attributum-szó, ez a berendezés jellemzőit írja le. Az attributum-szó egyes bit-jeinek jelentése a következő: 15. Ha karakteres periféria, akkor 1, különben 0. 14. Ha a DOS a perifériára elfogadja a 44-es funkcióhívás 3., 4. és 5. alparancsát, azaz támogatja az IOCTL-t (perifériát vezérlő információ), akkor 1, különben 0. 13. Nem IBM formátumú blokkos periféria esetén 1, IBM formátumú blokkos periféria esetén 0. 12. Fenntartott. 11. Ha az egység megérti az OPEN/CLOSE parancsot 1, különben 0. 10.-7. Fenntartott. 6. Ha az egység megérti a 3.2 és 3.3 parancsokat - támogatja a Get/Set Logical Device-t (lemez logikai/fizikai hozzárendelését) - akkor 1, különben 0. 5.-4. Fenntartott. 3. Ha a periféria a használt óraeszköz, akkor 1, különben 0. 2. Ha a periféria az aktuális NUL berendezés, - azaz erre tetszőleges output-ot kiküldhetünk anélkül, hogy bármi hatása lenne -, akkor 1. 1. Ha a periféria az aktuális konzol output, akkor 1, különben 0. 0. Ha a periféria az aktuális konzol input, akkor 1, különben 0.
6-7	Szó, amely annak a segédrutinnak a belépési címe, ami átveszi a perifériakezelő részére kiadott parancsblokkot (a parancsblokk leírása később következik).
8-9	Szó, amely a parancsot végrehajtó segédrutin belépési címe.
0A-11	8 byte, amely a berendezés nevét tartalmazza.

Megjegyzés:

1. A 44-es funkció segítségével a perifériára vezérlő karakterek küldhetők, illetőleg onnan vezérlő karakterek fogadhatók. Ezen a módon kezelhető például a baud rate, a stop bit, a változó formátumú rekord hossza stb. A funkció használatát részletesen ismerteti a funkcióhívásokat tartalmazó fejezet megfelelő szakasza.

2. A fejrész utolsó 8 byte-ja a névmező, amely karakteres perifériáknál egy maximum ilyen hosszú, a DOS szabályainak megfelelő nevet tartalmaz. Ha a név rövidebb, mint 8 karakter, akkor a mező elején (0A. azaz decimális 10. byte-on) kell kezdődnie, és a végén lévő szabad byte-okat szóközzel kell kitölteni.

   Blokkos perifériánál az első byte-nak az alegységek (unit) számát kell tartalmaznia. Ezek száma és a már lefoglalt egységek alapján rendel betűjelet a perifériákhoz a DOS. A továbbiakban az egységre a parancsokban az így kapott betűvel hivatkozhatunk.

   Amennyiben a névmező létező periférianevet tartalmaz, úgy a DOS a saját perifériakezelője helyett a felhasználóét fogja használni.

14.2.2 A perifériakezelő programokkal kapcsolatos tudnivalók összefoglalása és kiegészítése

Mint már említettük, a perifériakezelő program részei:

- a perifériakezelő fejléce,
- a parancsblokk átvevésére szolgáló segédrutin (ennek belépési címét a fejléc 6. és 7. byte-ja kell, hogy tartalmazza),
- a konkrét feladatot végrehajtó segédrutin (ennek belépési címét a fejléc 8. és 9. byte-ja kell, hogy tartalmazza).

Ha a CONFIG.SYS-ben elvégeztük az említett (DEVICE = file-név) javítást, akkor, új bekapcsolás vagy rendszertöltés után, a DOS betölti a file-t, amely tartalmazza a driverünket. Ezután minden file-számos file-megnyitásnál ellenőrzi, hogy a megnyitandó filenév nem egyezik-e valamely perifériakezelő névvel. Ha igen, akkor erre a file-ra való hivatkozások a DOS-ból kiváltják a driver-ünk megfelelő paranccsal való meghívásait. A driver meghívása két lépésből áll. Az elsőben a DOS felépít egy megfelelő parancsblokkot és meghívja a driver parancsblokk kezelő rutinját. E híváskor a DOS átadja a rutinnak ES:BX-ben az általa felépített fejléc címét. Ezen a területen a "parancskód"-nak nevezett byte-ban közli, hogy milyen feladat ellátását várja a driver-től, és kiegészíti a fejlécet az adott parancskódot jellemző, további parancsblokk részlettel.

A parancsblokk-átvevő segédrutint nekünk kell megírnunk. Minden esetre el kell tennünk a kapott fejléc-címet, és a DOS-tól kapott parancskód szerint kitöltjük a fejléc adatait. Az is a mi feladatunk, hogy a gép regisztereit a rutin kezdetén kimentsük, és mielőtt a DOS-nak visszaadjuk a vezérlést, visszatöltsük. Ezt a másik segédrutinnál is meg kell tennünk.

Lehetőség van arra is az első segédrutinnál, hogy a más gépeknél "interrupt driver" feladatoknak nevezett perifériakezelési módszereket beprogramozzunk, például sorba állíthatjuk az egyazon perifériára vonatkozó kéréseket stb.

Ha minden mentési, ki- és visszatöltési feladatot elvégeztünk, visszaadjuk a DOS-nak a vezérlést.

Erre a DOS azonnal meghívja ismét a handler-ünket, ezúttal a második segédrutin belépési címére adva a vezérlést. Most kell ellátnunk azt a feladatot, amit a DOS a parancskódban közölt velünk.

Mielőtt ismertetnénk a parancsblokk felépítését és a lehetséges parancskódokat, néhány dologra felhívjuk a figyelmet:

A DOS csak a regiszterek kimentésére biztosít veremterületet. Ha többre van szükségünk, azt magunknak kell lefoglalnunk és kezelnünk.

A DOS FAR CALL-lal hívja a rutinjainkat, így a rutinjainknak is ennek megfelelően kell visszatérniük. Mindkét segédrutint a fejléccel azonos szegmensben kell elhelyezni, és azoknak távoli RET utasítással (RETF azaz FAR RET) kell végződniük.

Ha több perifériakezelő található egy file-ban, ezeknek lehet közös segédrutinjuk, például a parancsblokk átvételére.

A parancsblokk-átadó segédrutin a fejléc címét a DOS-tól kapja, de a feladatot végrehajtó segédrutinnak már ő továbbítja, a feladatot kijelölő parancskóddal és egyéb információval együtt.

A perifériakezelő programoktól a DOS az itt következő feladatok ellátását várja el. (Az egyes feladatokhoz tartozó parancsblokkot a megfelelő feladatnál ismertetjük.)

DOS-tól kapott parancskód	**Az ellátandó feladat**
0	inicializálás
1	adathordozó csere megtörténtének ellenőrzése
2	BPB (Bios Paraméter Blokk, blokkos periférialeíró táblája) kitöltése (lásd a részletes ismertetést)
3	input/output vezérlő információ lekérdezése
4	olvasás perifériáról
5	karakteres periféria állapotának lekérdezése, az átviteli puffer tartalmának megőrzésével
6	karakteres periféria input puffer állapotának lekérdezése
7	karakteres periféria input puffer tartalmának törlése
8	írás perifériára
9	perifériára írás, írás utáni ellenőrzéssel
10	karakteres periféria output puffer állapotának ellenőrzése
11	karakteres periféria output puffer tartalmának törlése
12	perifériát vezérlő információk beállítása
13	periféria megnyitása (csak a 3.x verziókban, s ha a periféria attributum szavában a 11. bit magas)
14	periféria lezárása (csak a 3.x verziókban, s ha a periféria attributum szavában a 11. bit magas)
15	lemez cserélhetőségének lekérdezése (csak a 3.x verziókban, s ha a periféria attributum szavában a 11. bit magas)
16	írás perifériára míg foglalt nem lesz (csak a 3.10-es verzióktól)

19 vezérlő karaktersorozat elküldése (csak a 3.10-es verzióktól, s csak akkor, ha a periféria attributum szavában a 13. bit magas.)
23 lemez logikai-fizikai hozzárendelésének lekérdezése (csak a 3.2-es verzióktól, s ha a periféria attributum szavának 6. bitje magas)
24 lemez logikai-fizikai hozzárendelésének beállítása (csak a 3.2-es verzióktól, s ha a periféria attributum szavának 6. bitje magas)

A felhasználói perifériakezelő programot úgy kell megírni, hogy a fenti parancsokat fogadja a 14.2.4 részben részletezett formátumban, és a parancsok hatására megfelelően működjön.

14.2.3 A parancsblokk és fejlécének felépítése

A perifériakezelő program és a DOS operációs rendszer, mint említettük, egy úgynevezett parancsblokk segítségével kommunikál egymással. A parancsblokk első 13 byte-ja egységesen ugyanaz minden parancs esetében, ezt neveztük parancs fejlécnek (Request Header). Amikor a DOS meghívja a perifériakezelő program parancsblokk átvevő rutinját, akkor az ES:BX mutat a parancs fejléc első byte-jára. A rutinnak kell gondoskodni a mutató elmentéséről.

A fejléc szerkezete a következő:

Helyfoglalás (byte-ban) **Adattartalom**

1 A teljes parancsblokk hossza byte-okban kifejezve.
1 Blokkos periféria esetén a használandó alegység száma (például, ha a program 2 egységet kezel, e szám lehetséges értékei 0 és 1). (Karakteres periféria esetén ez a byte nem értelmezett.)
1 Parancskód. Ez fejezi ki a perifériakezelő program feladatát. A lehetséges értékeket a következő szakasz részletezi.
2 Állapotjelző szó. Tartalmát alább részletezzük.
8 Fenntartott terület a DOS számára.

Az állapotjelző szó tartalma:

bit **jelentés**

15 Hiba fellépésének jelzése. Értéke, ha nem történt hiba, 0, egyébként 1, és ekkor a hibakódot az állapotjelző szó alacsonyabb helyiértékű része tartalmazza, mint az majd következik.
14-10 A DOS számára fenntartott terület.
9 Úgynevezett foglaltsági bit. Bizonyos perifériakezelési feladatoknál a periféria foglaltságát vagy a háttértár cserélhetőségét jelzi. A következő szakaszban térünk erre vissza, a perifériakezelő programok feladatainak ismertetésénél.
8 A művelet befejezettségét jelző bit. Értéke akkor lesz 1, ha a művelet befejeződött, egyébként 0.

7-0 Hibakód. Ha a 15. bit 1, akkor a hiba kódját tartalmazzák, az alábbi táblázat szerint:

Hibakód	Hiba leírása
0	Íráskísérlet írásvédett lemezre.
1	Ismeretlen periféria.
2	A periféria nincs műveletre kész állapotban.
3	Ismeretlen fajta perifériakezelő parancs (lásd a következő szakaszt).
4	CRC hiba.
5	A parancsblokk hossza hibás.
6	Keresési (seek) hiba.
7	Ismeretlen adathordozó típus.
8	Nem találja az adott szektort.
9	A papír kifogyott a nyomtatóból.
0A	Íráshiba.
0B	Olvasási hiba.
0C	Nem azonosítható hiba.
0D	Fenntartott.
0E	Fenntartott.
0F	Nem megengedett lemezcsere.

14.2.4 A perifériakezelőt vezérlő parancsok részletes leírása

Inicializálás
Parancskód: 0
A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	Blokkos periféria esetén a handler által kezelt egységek, száma karakteres perifériáknál nincs értelmezve.
0E-11	A perifériakezelő utáni első szabad memóriacím értéke. Típusa: duplaszó.
12-15	Blokkos periféria esetén a BPB vektor, azaz a blokkos egységek leíró tábláit kijelölő címtáblázat címe, karakteres perifériánál nincs értelmezve. (A BPB ismertetése megtalálható a 14.1.1 részben.) Típusa: duplaszó.
16	A periféria egységszáma.

Feladata:

A DOS a perifériakezelő rutin betöltésekor egyszer hívja meg ezt a parancsot. A rutinnak be kell állítania az állapotjelző szót, a program által szabadon hagyott első memóriacímet, a kezelt egységek számát, valamint lemezes meghajtó esetén a BPB táblázat helyét megmutató címet. Ebben a programrészben kell a driver-nek minden olyan kezdeti feladatot ellátnia, amelyet a periféria megkíván.

Blokkos eszközök esetén a 12-15. byte-okon levő mutató a rutin meghívásakor a CONFIG.SYS file-ban levő DEVICE = utáni első karakterre mutat.Ez lehetővé teszi a paraméterek átvételét.

Adathordozó csere megtörténtének ellenőrzése
Parancskód: 1
A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	Az adathordozó típusa - a DOS adja meg. A típusleíró byte-ot a 2. parancsnál ismertetjük.
0E	Csere állapotának jelzése: -1 az adathordozót cserélték 0 a csere ténye nem megállapítható 1 csere nem volt.
0F-12	Csere esetén az adathordozó kötetazonosítójára (volume identifier) mutató duplaszó.

Feladata:

A parancs hatására meg kell állapítani, hogy történt-e a kérdezett egységben csere. A lekérdezés eredménye a következő 4 lehetőség valamelyike:

- nem történt csere,
- csere történt,
- a csere nem megállapítható,
- hiba lépett fel.

A megállapított eredménynek megfelelően be kell állítani a fejlécben a visszatérési kódot, a csere állapotát jelző byte-ot, továbbá csere esetén az új adathordozó kötetazonosítójára mutató pointert.

Megjegyzés:

1. Ha a DOS csere esetén megállapítja, hogy az hibás, akkor 0F kódú hibát jelez a kérdéses egységre.
2. Ha csere történt, de az adathordozó nem tartalmazott kötetazonosítót, akkor az erre mutató pointernek (0F-12) egy "NO NAME",0 string-re kell mutatnia.
3. A DOS által használt adathordozók típusai a következő, a 2-es parancsnál találhatók.
4. A DOS 3.2-esnél régebbi verziói az adathordozó típus byte-ját használták a lemez struktúrájának azonosítására. Ha csere történt, de a visszaadott BPB típusleíró byte-ja ugyanaz volt, akkor a DOS azt tételezte fel, hogy a cserélt lemez felépítése ugyanaz.

BPB (BIOS Parameter Blokk) kitöltése
Parancskód: 2
A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	Az adathordozó típusa. A programozó adja meg. (Lásd a megjegyzést.)
0E-11	A DOS által a handler részére átengedett puffer címe (duplaszó).
12-15	A perifériakezelő által összeállított BPB (egység leíró tábla) címe (duplaszó).

Feladata:

Az adathordozón található boot rekord alapján össze kell állítania az egységben található adathordozó a valóságnak megfelelő típus leíró byte-ját, és az egység leíró táblát. A típusleíró byte-ot és az egységleíró tábla címét a perifériakezelő a parancsblokkban visszaadja a DOS-nak.

Megjegyzés:

Az adathordozó típusleíró byte megtalálható a lemezeken a FAT 0. byte-jaként, és felépítése a következő:

7-3. bitek	Értéke kötelezően 1.
2. bit	Ha az adathordozó cserélhető, akkor 1, különben 0.
1. bit	Ha az adathordozó 8 szektoros, akkor 1, különben 0.
0. bit	Ha kétoldalas a lemez, akkor 1, különben 0.

A fenti értelmezés nem hagyott azonosító bit-et a lemezek minden jellemzőjére (például átmérő, felírási sűrűség). A létező azonosító byte-ok a fejezet elején találhatók meg.

Periféria vezérlő input/output parancsok
Parancskód: 3,4,8,9,12,16
A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	Az adathordozó típusa - a DOS adja meg.
0E-11	Duplaszó, amely az átviteli puffer címét tartalmazza.
12-13	Az átviendő byte-ok, illetve szektorok számát tartalmazza. A művelet befejezése után a periféria-kezelőnek be kell írnia a ténylegesen átvitt egységek számát.
14-15	Szó, amely blokkos eszköz esetén a kezdő szektor sorszámát tartalmazza.
16-19	Duplaszó, amelyben 0FH hibakódú hiba (nem megengedett lemezcsere) esetén be kell állítani annak a karaktersorozatnak a címét, amely a helyes kötetazonosítót tartalmazza.

Input/output vezérlőinformáció lekérdezése.
Parancskód: 3
Feladata:

A perifériakezelő programnak át kell adnia a parancsblokk adatainak megfelelő beállítása segítségével a periféria vonatkozó vezérlő információt. További információk találhatók erre vonatkozóan a 44-es funkció leírásánál.

Olvasás a perifériáról
Parancskód: 4
Feladata:

A parancsblokkban megadott számú byte-ot (karakteres periféria), illetőleg szektort (blokkos periféria esetén) kell a handler-nek beolvasnia a perifériáról.

Írás a perifériára
Parancskód: 8
Feladata:

A feladat ugyanaz, mint a 4. parancsnál, csak írni kell.

Perifériára írás, írás utáni ellenőrzéssel
Parancskód: 9
Feladata:

Ugyanaz, mint az előző parancs, de az írás sikerességét utólag le kell ellenőrizni és ennek megfelelő értékű állapotjelző szót kell a parancsblokk fejléc részében visszaadni.

A perifériát vezérlő információk beállítása
Parancskód: 12
Feladata:

A perifériakezelő program átveszi a 44-es funkciónál leírtak szerint a kívánt vezérlő információkat, és a továbbiakban ennek megfelelően kell működnie.

Írás a perifériára, míg foglalt
Parancskód: 16
Feladata:

A 16. parancs biztosítja, hogy ki lehessen használni a legtöbb nyomtatónál rendelkezésre álló pufferolási lehetőséget. A pufferolási lehetőségek ismeretében nem kell megvárni, míg a nyomtató jelzi, hogy kész, hanem rögtön vissza lehet adni a vezérlést a DOS-nak. (Ha az attribútum szóban beállítottuk, hogy a periféria megérti a 3.2-es parancsokat, csak abban az esetben használhatjuk ezt a parancsot.)

Karakteres periféria olvasása az átviteli puffer tartalmának megőrzésével
Parancskód: 5

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	A perifériáról olvasott byte-ot ide teszi a handler.

Feladata:

A handler-nek a karakteres periféria pufferének állapotát kell a PC-DOS felé visszajeleznie a következő módon:

- ha a puffer üres, akkor a parancsblokkban az állapotjelző szó foglaltsági bitjét (9. bit) nullára kell állítani,
- ha a puffer adatot tartalmaz, akkor a jelzett bitet 1-re kell állítani, továbbá a pufferből a karaktert be kell rakni a parancsblokkba. A puffer tartalma nem törlődik.

Karakteres periféria input puffer állapotának lekérdezése
Parancskód: 6

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.

Feladata:
A parancs megegyezik az 5-ös paranccsal. Az egyetlen eltérés, hogy a parancsblokkban csak az állapotot kell visszaadni, a pufferban lévő karaktert nem.

Megjegyzés:
Azok a perifériák, amelyeknek nincs pufferük, az 5. és a 6. parancsnál is 0 foglaltsági bitet kell hogy visszaadjanak, hogy a DOS ne várakozzék feleslegesen.

Karakteres periféria input puffer tartalmának törlése
Parancskód: 7

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.

Feladata:
A perifériakezelőnek törölnie kell az input pufferben található valamennyi karaktert, törölnie kell az összes, a perifériára vonatkozó kérést, amelyet nyilvántartott, és be kell állítania a fejlécben az állapotjelző szót.

Karakteres periféria output puffer állapotának ellenőrzése
Parancskód: 10

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.

Feladata:
Ha a handler kész a következő karakter feldolgozására, akkor az állapotjelző szó foglaltsági (9.) bitjét nullára kell állítania, ellenkező esetben 1-re.

Karakteres periféria output puffer tartalmának törlése.
Parancskód: 11

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.

Feladata:
A handler-nek törölnie kell az output puffer teljes tartalmát, és be kell állítania az állapotjelző szót.

Adathordozó cserélhetőségének lekérdezése
Parancskód: 15

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.

Feladata:
A parancsot csak a DOS 3.xx-es változatai hívják, s azok is csak akkor, ha a periféria-kezelő attributum szavában beállítottuk az OPEN/CLOSE/RM bitet. A parancs végrehajtása után a periféria-kezelőnek jeleznie kell, hogy az adathordozó cserélhető-e vagy sem. Az információt az állapotjelző szó foglaltságot jelző bitjében kell visszadni. Ha magas (1), akkor a lemez nem cserélhető, ha 0, akkor igen.

Vezérlő karaktersorozat átvitele a perifériára
Parancskód: 19

A parancsblokk felépítése:

Byte	Jelentés
0-0C	A parancsblokk fejléce.
0D	Byte: kategória kód.
0E	Byte: függvény kód.
0F-10	Szó: (SI) tartalma.
11-12	Szó: (DI) tartalma.
13-16	Duplaszó: az adatpuffer kezdőcímére mutat.

Feladata:
A parancs a 44-es funkció 2,3,4,5 alfunkcióit hajtja végre az adott perifériára. Az egyes paraméterek leírása ott található meg.

File megnyitása
Parancskód: 13

File lezárása
Parancskód: 14

A parancsblokk felépítése:

Byte Jelentés
0-0C A parancsblokk fejléce.

A DOS 3.xx-es verziói, ha a beállítottuk az OPEN/CLOSE/RM bitet a periféria attributum szavában, jelzik a periféria kezelőnek, ha a periférián megnyitottak/lezártak egy file-t. Lehetőség van ilyenkor valamilyen tevékenység elvégzésére (pl. egy y inicializáló karaktersorozat elküldésére a nyomtatóra), vagy hibát jelezni, ha egynél több file nem lehet nyitva a periférián.
A CON, AUX és PRN perifériák mindig nyitottak (a hozzájuk rendelt file-számok: 0,1,2,3,4)

Logikai-fizikai hozzárendelés lekérdezése
Parancskód: 23

Logikai-fizikai hozzárendelés beállítása
Parancskód: 24

A parancskód felépítése:

Byte Jelentés
0-0C A parancsblokk fejléce.

Feladata:

A DOS ezt a funkciót csak akkor hívja, ha a periféria attributum szavában a 3.2-es bitet beállítottuk. Ebben az esetben ugyanazt a fizikai eszközt több logikai eszköz is birtokolhatja. A két parancs ezt a hozzárendelést kérdezi le, illetve állítja be.

A DOS a parancsblokk fejlécének az egységszámot tartalmazó részében adja át a logikai egységszámot és ott kéri vissza azt is, hogy kié volt a meghajtó utoljára.

14.2.5 Javaslat gyakorlati példára

Az IBM PC-k kedvelt tartozéka, az AT-k szerves része a "Real Time Clock", azaz a hálózatfüggetlen óra. Ezt a kártyát az átlag felhasználó a TIME és a DATE parancsokkal éri el. Ez speciális perifériaként funkcionál.

Amint azt a perifériakezelő fejléc attributum szavának leírásánál mondottuk, az attributum szó 3-as bitje jelzi, ha a periféria az óra. Az óra kezelésére a DOS-ban levő driver teljesen hasonlóan működik, mint a normál karakteres perifériáké. Amennyiben olvasni próbálunk a perifériáról, vagy írni akarunk rá, úgy 6 byte kerül átvitelre. Az első két byte egy szó, amely folyamatosan számolja a napokat 1980. január 1-től kezdve. A harmadik byte a perceket, a negyedik az órákat, az ötödik a századmásodperceket, végül a hatodik a másodperceket tartalmazza.

A megfelelő DOS funkciók segítségével nyissuk meg (3D funkció) a CLOCK$ nevű file-t, és olvassuk le róla (3F funkció) a fenti hat byte-ot. E funkció hívásakor a DOS az előző szakaszban ismertetett parancsok segítségével látja el a funkcióhívásokkal megadott feladatot.

Végül azoknak, akik a gyakorlatban meg akarják ismerni egy felhasználói perifériakezelő rutin szerkezetét, javasoljuk, hogy a DEBUG segítségével vizsgálják meg az ANSI.SYS driver-t. Mint ismeretes, az ANSI.SYS biztosítja, hogy escape szekvenciákkal lehessen a képernyőre írást irányítani, Azért javasoljuk ezt, mert a DOS 2.00 vagy későbbi változatokhoz az ANSI.SYS file biztos, hogy rendelkezésre áll.

A fenti módszernél lényegesen egyszerűbb, ha a 3.00-ás verziótól kezdődően a DOS-hoz adott VDISK.LST nevű file-t tanulmányozzák át, amelyik a VDISK.SYS driver assembly listáját tartalmazza.

15. A DOS memóriakezelése, munkaterületei és vezérlőblokkjai

Ez a fejezet a memória felosztását és kezelését, a DOS-nak a programokról és a file-okról nyilvántartott információit és ezek felhasználási lehetőségét tekinti át.

A memóriatérkép ismertetése után részletezzük a DOS adatterületeit és az alkalmazói programokat kiszolgáló táblázatokat: a PSP-t (Program Segment Prefix) és a programkörnyezetet (environment). Ezt követően a memória kezelésével és nyilvántartásával foglalkozunk. A fejezet második része a perifériakezeléssel kapcsolatos táblázatokat (file leíró blokk, lemezátviteli terület, felhasználói perifériakezelők vezérlőblokkja) foglalja össze.

15.1 A DOS memóriatérképe

A memóriatérkép megmutatja, hogy a számítógép memóriájában milyen sorrendben és milyen címeken helyezkednek el a BIOS (Basic I/O System) és a DOS elemei, valamint a felhasználói programok.

Memóriacím (szegmens:offset)	**Terület megnevezése**
0000: 0000-03FF	megszakítási vektor tábla
0000-007F	BIOS megszakítások (INT 0-1F)
0080-00FF	DOS megszakítások (20-3F)
0100-017F	fenntartott (40-5F)
0180-019F	megszakítások felhasználói programok részére (60-67)
01A0-01FF	nem használt megszakítások (68-7F)
0200-0217	foglalt a BASIC részére (80-85)
0218-03C3	BASIC interpreter által a programok futása alatt használt megszakítások (85-0F0)
03C4-03FF	nem használt megszakítások (0F1-0FF)
0040: 0000-00FF	BIOS adatterület
0050: 0000-00FF	DOS adatterület
0000	képernyőprintelés (hardcopy) jelző; ha tartalma: 0 - képernyőprintelés inaktív vagy sikeresen végetért 1 - képernyőprintelés folyamatban 255 - hiba lépett fel a képernyőprintelés alatt
0001	a BASIC használja
0004	egy meghajtós üzemmód jelző, ha tartalma: 0 - az A meghajtót használták utoljára 1 - a B meghajtót használták utoljára
0050: 0010-0021	a BASIC használja
0022-002F	adatterület lemezmeghajtók inicializálásához
0030-0033	adatterület a MODE parancs részére

XXXX:0000-	a ROM I/O rutinokra épülő alapszintű perifériakezelő (IBMBIO. COM)
YYYY:0000-	a DOS megszakítást kezelő és funkcióhívási rutinjai (IBMDOS. COM)
	DOS pufferek, vezérlő területek és a felhasználói perifériakezelők
ZZZZ:0000-	a parancsfeldolgozó (COMMAND.COM) állandó részei, valamint a DOS 22, 23, 24-es megszakításkezelő rutinjai
VVVV:0000-	külső parancsok vagy segédprogramok (.COM vagy .EXE file-ok) veremtár a .COM programok részére (decimális 256 byte)
WWWW:0000-	a parancsfeldolgozó átmenetileg tárolt részei, azaz parancs értelmező, belső parancsok, batch feldolgozó, külső parancs betöltő

A memóriatérképben szereplő XXXX, YYYY, ZZZZ, VVVV, WWWW szegmenscímek értéke a DOS megvalósításától függően változhat, de mindenképpen egymást követő tárterületeket jelölnek ki.

15.2 A programszegmens prefix (PSP)

Külső parancs vagy felhasználói program indításakor az operációs rendszer a legalacsonyabb című szabad tárterületet használja fel. Az így felhasznált memória a programszegmens. A programszegmens első 256 (hexadecimálisan 100) byte-ján az operációs rendszer egy vezérlőblokkot, az úgynevezett PSP-t (Program Segment Prefix-et) hoz létre. A PSP-t követően helyezkedik el az indítandó program futásra alkalmassá tett kódja. (A 'COM' kiterjesztésű programok tárfüggetlen kódot tartalmaznak, azaz tetszőleges helyre betöltve hibátlanul működnek. Az 'EXE' kiterjesztésű programok ellenben olyan utasításokat is tartalmaznak, amelyek függnek a program kezdőcímétől, ezért a betöltéskor ezeket az utasításokat módosítani kell. Ezt a módosítást az operációs rendszer végzi el. Ez a művelet a relokálás (programáthelyezés, ezt a műveletet az EXE fejléc leírásánál ismertetjük majd).

Az operációs rendszer programindításkor a programszegmens számára lefoglalja a felhasználható teljes tárterületet. Amennyiben a program ekkora területet nem igényel, módosíthatja a szegmens méretét. Ha a program befejezte a futását, a DOS-hoz visszatérve a visszatérés módja szerint vagy megszűnik a programszegmens, vagy megmarad (rezidens kilépés). Utóbbi esetben a későbbiek során a rezidens program újabb betöltés nélkül elindulhat. Ilyen esetekben a gép memóriájában egyidejűleg több programszegmens követi egymást. Normál programbefejezés elérhető INT 20 hívással, vagy a 0-ás, vagy a 4C-s funkcióhívással, továbbá a PSP 0. byte-jára való ugrással (JMP 0). Megjegyezzük, hogy az utóbbi esetben, valamint INT 20 használatakor gondoskodni kell arról, hogy a CS szegmensregiszter a PSP-re mutasson. Ellenkező esetben (azaz ha a PSP nem a CS:0000 címen található), a programbefejezés után nem fog a DOS bejelentkezni, a további munkához rendszertöltés szükséges. Ha azt akarjuk, hogy a program befejezése után is megmaradjon a programszegmens, akkor INT 27-tel vagy a 31-es funkcióhívással kell lezárni.

15.2.1 A PSP felépítése

Mint említettük, a gép memóriájában egyidejűleg több PSP is bent lehet. A tárkezeléshez, a program indításához és befejezéséhez, valamint betöltéséhez,továbbá az operációs rendszerrel való kapcsolathoz szükséges információk a PSP-ben találhatók. Ebből következik, hogy a PSP bizonyos adatait a program felhasználhatja, egyeseket módosíthat is, másokat azonban nem változtathat meg. A 0-5C közötti területet nem szabad átírni.

A PSP a következőket tartalmazza:

Cím byte	Tartalom
0-1	A 2 byte tartalma: CD 20. Ez egy INT 20 utasítás. A program erre a címre ugorva befejezheti a futását. (A programbefejezési lehetőségeket lásd az interrupt-ok ismertetésénél.)
2-3	Memória felső határa. Az első nemlétező memóriacím paragrafusban kifejezve. Példánkban ez 3D80. Ez decimálisra átszámolva, és figyelembe véve, hogy egy paragrafus 16 byte, a gép memóriájára 247.808 byte-ot ad.
4	Fenntartott
5-9	A DOS funkciókkal kapcsolatos LONG CALL utasítás.
0A-0B	Program befejezési cím offsetje.
0C-0D	Program befejezési cím szegmenscíme.
0E-0F	< Ctrl-Break > kilépési cím offsetje.
10-11	< Ctrl-Break > kilépési cím szegmenscíme.
12-13	Hibakezelő rutin cím offsetje.
14-15	Hibakezelő rutin cím szegmenscíme.
16-2B	DOS részére foglalt terület.
2C-2D	A programkörnyezet szegmenscíme.
2E-4F	DOS részére foglalt terület.
50-52	Ez a három byte egy, a DOS funkciókat (példánkban CD 21 CB, ami éppen INT21 RET FAR) meghívó utasítást tartalmaz. Így a program a 21-es interrupt-on kívül úgy is hívhat DOS funkciót, hogy AH-ban beállítja a kívánt funkció számát, és FAR CALL szubrutinhívást ad ki erre a címre.
53-5B	DOS részére foglalt terület.
5C-6B	Normál nem megnyitott file-leíró blokk (FCB1).
6C-7F	Második normál nem megnyitott file leíró blokk (FCB2). Az FCB1-et és az FCB2-t a DOS alakítja ki. Ha az adott PSP-t létrehozó DOS parancs file-t vagy file-okat is megad, akkor e paraméterekből kiveszi a file-okra vonatkozó információt és ide beírja, amint a példában is látni fogjuk. Ha a parancs nem ad meg file-t, akkor a meghajtó az aktuális lesz (0), a file-azonosító meg space-szel lesz feltöltve. Ha az FCB1-et megnyitjuk, akkor az FCB2 felülíródik.
80-FF	Lemez átviteli terület (DTA). A parancsfeldolgozó ezt jelöli ki DTA-nak és a 81. byte-tól kezdődően itt tárolja a konzolról beolvasott sor parancsot követő részét, a program indítási paramétert. A program indítási paraméter hossza a 80-as címen található.

A fentieket egyszerűen megnézhetjük a DEBUG utasítás segítségével.

Adjuk ki a DEBUG COMP.COM A:XXX B:YYY parancsot.

A DEBUG utasítás a betöltődése után be fogja tölteni az indítási paraméterén található file-t, azaz COMP.COM-ot. A betöltés után a -DO.100 parancs segítségével megnézhetjük a PSP-t, ami COMP A:XXX B:YYY parancshoz tartozik:

```
0DA0:0000 CD 20 80 3D 00 9A F0 FF-0D F0 42 02 99 0A 70 02  M .=..p..pB...p.
0DA0:0010 99 0A A2 03 34 06 99 0A-01 01 01 00 02 FF FF FF  ..".4...........
0DA0:0020 FF FF FF FF FF FF FF FF-FF FF FF FF 9B 0D 49 2A  ..............I*
0DA0:0030 99 0A 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0dA0:0040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:0050 CD 21 CB 00 00 00 00 00-00 00 00 00 01 58 58 58  M!K..........XXX
0DA0:0060 20 20 20 20 20 20 20 20-00 00 00 00 02 59 59 59            .....YYY
0DA0:0070 20 20 20 20 20 20 20 20-00 00 00 00 34 06 00 00            ....4...
0DA0:0080 0C 20 61 3A 78 78 78 20-62 3A 79 79 79 0D 78 78  . a:xxx b:yyy.xx
0DA0:0090 20 62 3A 79 79 79 0D 00-00 00 00 00 00 00 00 00   b:yyy..........
0DA0:00A0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:00B0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:00C0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:00D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:00E0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
0DA0:00F0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ................
```

A 89 címen lévő 0D a parancssort lezáró kocsivissza, az ezt követő betűk pedig mutatják, hogy a DEBUG betöltésekor a parancssor hosszabb volt.

15.2.2 A PSP egy program indításakor

A programok indításakor a DOS a következő feladatokat látja el:

1. PSP létrehozása, memóriafoglalás, a PSP 5C-től és 6C-től kezdődően egy-egy FCB kitöltése, valamint a program indítási paramétereinek tárolása a PSP + 81-től kezdődően. A DOS a programnevet követő space-től a soremelésig található karaktersorozatot (string-et) indítási paraméterként kezeli és a jelzett helyre másolja. Nem kerülnek azonban ide a parancssorból a <, >, > >, valamint P| jelekkel kezdődő szavak, hanem ezeknek megfelelően a parancsfeldolgozó elvégzi az input, illetőleg output átrendeléseket, vagy gondoskodik a parancsok láncolásáról.

2. A programkód elhelyezése a PSP utáni területre és EXE programok esetén a relokáció elvégzése.

3. A DS, ES, SS és SP regiszterek beállítása. (A DS és ES mind a COM, mind az EXE kiterjesztésű programoknál a programszegmens elejére mutat. Az SS és SP regiszterek értékei EXE programoknál az EXE-fejléc szerint alakulnak, míg COM programoknál SS = DS és SP = 0FFFE).

4. A vezérlés átadása a kijelölt programkezdetre, ami COM programok esetén CS = DS:100, azaz a PSP-t követő program fizikai kezdete. EXE programok esetében a CS és az IP értéke az EXE-fejléc alapján állítódik be. Emlékeztetőül:

SS = Stack Segment regiszter
SP = Stack Pointer, azaz a veremtár címe SS:SP
DS = Data Segment
CS = Code Segment - ez jelzi az utasítások címének a szegmensét
IP = Instruction Pointer - ez mutatja a következő utasítás címének a helyét a szegmensen belül (cím offset-je), azaz a következő utasítás címe: CS:IP.

15.3 A programkörnyezet és a memórianyilvántartás

A fejezet eddigi részében megismerhettük a programszegmens felépítését, a program betöltésének és indításának folyamatát. Az ott leírtak szerves folytatását képezi a következő alfejezet, ugyanis az előzőekben csak érintőlegesen említett fogalmak részletezésére kerül itt sor. A PSP-ben található mutató adja meg mind a programkörnyezet, mind pedig a memórianyilvántartó blokk helyét. Ezek a területek a felhasználók számára elsődlegesen érdektelenek, de bonyolultabb struktúrájú programok létrehozásakor ezek ismerete hasznos lehet. Az, hogy ismeretük elsődlegesen érdektelen, azt jelenti, hogy általános igényű programokat írva, illetőleg ilyeneket használva nem fogunk olyan problémákkal találkozni, amelyek megértéséhez, elkerüléséhez vagy megoldásához foglalkozni kellene a memórianyilvántartással vagy a programkörnyezettel. Speciális igények esetén azonban más a helyzet, amint azt példán is be fogjuk mutatni.

15.3.1 Programkörnyezet

A programkörnyezet egy maximum 32 Kbyte hosszú karaktersorozat, amely a következő formában tartalmazza a DOS állapotát, területeit jelző és a program számára információkat adó paramétereket:

KULCSSZÓ = paraméter 0 . . . . . KULCSSZÓ = paraméter 0 0

Mind a kulcsszavak, mind pedig a paraméterek ASCII karakterekből állnak. A kulcsszóhoz tartozó paraméter végét egy nullát tartalmazó byte jelzi. A programkörnyezet végét az utolsó paramétert lezáró nullát követően egy további nullát tartalmazó byte mutatja.

A programkörnyezet mindig tartalmazza a következő kulcsszavakat:

COMSPEC - a hozzá tartozó paraméterérték egy elérési utat tartalmaz, amely kijelöli az aktuális COMMAND.COM helyét. Felhasználható például a COMMAND.COM programból való elindítására.

PATH - a hozzá tartozó paraméterérték elérési utat (ill. pontosvesszővel elválasztva elérési utakat) tartalmazza. Ezt a PATH parancs állítja be és a parancsfeldolgozó számára jelöli ki az elérhető tartalomjegyzéket, illetőleg tartalomjegyzékeket. Részletesebb információk a PATH parancs leírásánál találhatók.

PROMPT - a hozzá tartozó karaktersorozat vezérli a parancsfeldolgozó kijelentkezésének formáját. A karaktersorozat megváltoztatását, illetőleg lecserélését a PROMPT parancs végzi. (Részletesebb információk a PROMPT parancs leírásánál találhatók.)

A fentieken kívül tetszőleges kulcsszó és paraméter bevihető a programkörnyezetbe a SET parancs segítségével. Ezt felhasználva olyan felhasználói program írható, amely a programkörnyezetben szereplő speciális kulcsszavakat fel tudja használni. Például egy grafikusan nyomtató program különféle típusú nyomtatókat ismer. Amennyiben a programkörnyezet nem tartalmazza a PRINTER= kulcsszót és ezt követően egy a program által ismert típusjelzést, úgy a rendszer megkérdezi a printer típusát. Ellenkező esetben a programkörnyezetben megadott típusú printert fog feltételezni. Ez a megoldás kényelmes a felhasználó számára, hiszen csak egyszer kell az AUTOEXEC.BAT file-ba beírnia egy SET PRINTER = típus parancsot, és ettől kezdve a rendszertöltés automatikusan közli a printer típusát az említett programmal.

Az aktuális programkörnyezet teljes tartalma az operációs rendszerből a paraméter nélkül megadott SET paranccsal kérdezhető le. Maximális mérete 32 Kbyte lehet. Vigyázni kell azonban, ha a tárban rezidens program található. Ekkor ugyanis a programkörnyezet nem haladhatja meg a rezidens program betöltésekor meglévő méretét. A méret túllépésének megkísérlése "Out of environment space" hibaüzenetet eredményez. Azaz a fenti üzenetet kapjuk, ha rezidens program mellett a PROMPT, a PATH vagy a SET paranccsal a korábbinál hosszabb paramétert, illetőleg a SET segítségével új kulcsszót akarunk megadni.

Példa:

Egy PATH \ parancs kiadása után, ha PRINT paranccsal kinyomtatunk egy file-t, akkor a PRINT már benne marad a memóriában újabb rendszertöltésig.

Ha ekkor kiadunk egy PATH A:\ parancsot, ami az "A" egység főkönyvtárát jelölné ki, ezt már a gép nem tudja elfogadni, a fenti hibaüzenetet kapjuk.

15.3.2 A memórianyilvántartás

A DOS 4B funkciója, amely betölti és elindítja a felhasználói programokat, a betöltés során a program rendelkezésére bocsájtja a lehetséges maximális memóriát. Ezért, ha egy program ilyen módon akar további programot indítani (overlay), gondoskodnia kell arról, hogy az általa nem használt területből felszabadítson annyit, amennyi az új program betöltéséhez elég. Ha az így betöltött és elindított program befejezte futását, akkor a hívó program újból lefoglalhatja saját számára ezt a területet. A memóriakezelés céljára három funkcióhívás szolgál:

A 48-as hívással tetszőleges méretű memóriablokkot lehet lefoglalni.
A 49-es hívással egy lefoglalt blokk felszabadítható.
A 4A-s hívással egy memóriablokk mérete módosítható.

Memóriablokk alatt az operációs rendszer egy egybefüggő memóriaterületet ért, melynek a helyét és méretét a rendszer tartja nyilván. Az indításkori memóriablokk méretét

közvetve meg lehet kapni, mivel a PSP 2. byte-ján kezdődő szó megadja a legmagasabb szegmensként címezhető területet. (Amint a címzéseknél ismertettük, a memóriában minden cím egy szegmens és egy eltolási érték összegeként áll elő, ahol a szegmens 16 byte-os egységekben, azaz paragrafusokban van. Ez azt jelenti, hogy a cím a (szegmens*16) + eltolás képlet alapján számítódik.)

Amint azt a funkciók leírásánál ismertetjük is, a 48-as funkció, ha akkora memóriát kérünk, amennyi már nincs, visszaadja paragrafusban számítva a legnagyobb rendelkezésre álló memóriablokk méretét. Az is egy megoldás, hogy eleve a program mérete szerint módosítjuk a memóriablokk méretét.

A PC DOS Technical Reference nem közli a memóriakezelés minden részletét, de más forrásokból és ilyen problémákat megoldó programokból (pl. DEBUG, BASICA) a következők állapíthatók meg:

A DOS betöltő, amikor létrehoz egy memóriablokkot, a megelőző paragrafuson (eggyel kisebb szegmensen) egy memória leíró blokkot (Memory Control Block = MCB) hoz létre. Az MCB első byte-ja jelző. Ennek tartalma 4D, ha további memóriablokkok vannak még. Az utolsó memóriablokkot 5A jelzi. Ezt két szó követi. Az első megadja az MCB által foglalt memória kezdetének szegmenscímét (paragrafus). A második a kezelt memóriablokk méretét adja paragrafusban. Utolsó MCB esetén ez az érték a maradék szabad memóriaterület nagyságát adja. Nem utolsó memóriablokk esetén a következő memóriablokk szegmenscímét megkapjuk, ha a két szó tartalmát összeadjuk és az eredményt eggyel növeljük.

Táblázatosan az MCB felépítése:

1. byte	Jelző, tartalma 4D = nem utolsó blokk 5A = utolsó blokk
2-3. byte	Memóriablokk kezdetére mutató érték
4-5. byte	A nyilvántartott blokk mérete paragrafusban.

A fentiek ismeretében felvetődik a kérdés: hogy viszonyulnak a fenti funkcióhívások és az MCB egymáshoz?

A helyzet a következő: Ha 48-as funkcióhívással létrehozunk, vagy 4A hívással méretében módosítunk egy memóriablokkot, akkor új MCB keletkezik, és a korábbiban a méret átíródik.

Hasonlóan, 49-es hívás esetén, a megszüntetett blokk MCB-je kiiktatódik a láncból. (Az eddigiekből látható ugyanis, hogy a memóriablokkok nyilvántartását az MCB-k sorozata végzi, ahol a megelőző implicite tartalmazza az őt követő címét.)

Amennyiben a 49-es vagy a 4A funkcióhívásban az ES nem MCB-re mutató címet tartalmaz, úgy a hívás 9-es hibát jelez. (Carry magas és AL = 9)

Összefoglalva:

1. Minden programszegmens méretét a PSP-t megelőző szegmensen található memória leíró blokk (MCB) tartja nyilván.

2. A DOS funkcióhívás memóriakezelése MCB-k tartalmának módosításával, illetőleg ezek létrehozásával valósul meg.

3. Egy memóriablokk - ez lehet programszegmens vagy önállóan létrehozott memóriablokk - mindig akkora méretű, amekkorának lefoglaltuk vagy módosítottuk.

4. Ha kezelni akarunk egy memóriablokkot, akkor mindig a blokk kezdetének szegmenscímével kell azonosítanunk. (Ezt az értéket kell az ES regiszterbe tenni a 49-es és a 4A funkcióhívás előtt, illetőleg ezt az értéket kapjuk vissza az AX regiszterben, ha a 48-as funkcióhívás eredményes volt.)

A memóriaallokáció első szegmensének (a lánc első elemének) címét a PSP 16-17. bytejából olvashatjuk ki. Ezután a láncon végighaladva megtudhatjuk azt is, milyen tárrezidens programok kerülnek a memóriába.

5. Amennyiben programhiba miatt valamelyik MCB felülíródik, a program terminálásakor a DOS "Memory allocation error - System halted" hibaüzenetet ír ki. Újra kell tölteni a DOS-t (boot)!

Végül megjegyezzük, hogy ha egy program hibakezelése nem megfelelő, akkor előfordulhat, hogy a DOS-hoz kerül a vezérlés (pl. lemezműveleti hibára abort választ adva) anélkül, hogy a program területe felszabadulna. Kisebb memóriájú gépeknél, vagy ha ez egymás után többször ismétlődik, elfogyhat a szabad memória. Ilyenkor a rendszer egy újabb program indítására "No memory" hibaüzenetet ad.

Az eddigiekből látható, hogy a memóriakezelés meglehetősen nehézkes, ezért ilyenkor a legegyszerűbb megoldás a memória felszabadítására az új rendszertöltés. <Ctrl-Alt-Del>

15.4 A file-kezelés és munkaterületeinek szerkezete

A PC DOS operációs rendszer legkorábbi változata egyetlen, a CP/M operációs rendszerhez hasonló file-kezelési eljárást ismert. Ennek a lényege, hogy minden fileművelethez egy, a művelet minden fontos paraméterét tartalmazó, meghatározott szerkezetű terület, az úgynevezett file leíró blokk (FCB), valamint egy, az opercációs rendszer által nyilvántartott munkaterület, a lemez átviteli terület (DTA) szükséges. Ezeket a felhasználó a memória tetszőleges helyén kijelölheti. Az operációs rendszer a file létrehozásához, megnyitásához és lezárásához csak a file leíró blokkot, olvasási és írási műveletekhez a lemez átviteli területet is igényli. Az írási és olvasási műveletek mindig az FCB-ben megadott méretű blokkokban történnek. A file munka közbeni állapotáról információkat a felhasználó a file leíró blokkból szerezhet. A továbbiakban az eljárást, illetőleg a hozzá tartozó funkcióhívásokat a file leíró blokk neve alapján FCB-orientált eljárásnak, illetőleg funkcióhívásoknak fogjuk nevezni.

A 2.00 és magasabb verziószámú DOS operációs rendszerek többek között egy új filekezelési móddal is kiegészültek. Ennél a módszernél a felhasználó minden átviteli műveletnél megadja az átvitelre kerülő adatok helyét és mennyiségét, továbbá a file azonosítása nem a file leíró blokkal, hanem egy sorszámmal, az úgynevezett file-számmal történik. Ezt az operációs rendszer közli a felhasználóval a file megnyitásakor. A file-név

megadása így csak a file megnyitásakor szükséges. Ezt az eljárást file-számos file-kezelésnek fogjuk hívni.

A fejezet további részei először kiemelten ismertetik a file-kezelési eljárások fogalmait, majd külön szakasz foglalkozik az egyik és a másik eljárás használatával is. Bár ez a feldolgozás a legelső olvasáskor egy kicsit több munkát igényel az olvasótól, mert az egyes fogalmak részleteihez vissza kell lapoznia, viszont a későbbiekben, amikor csak egy-egy részletre van szükség, lényegesen kényelmesebbé teszi a könyv használatát.

15.4.1 File leíró blokk (FCB)

A file leíró blokk (File Control Blokk = FCB) az a terület, ahol az FCB-orientált file-kezelés során a file valamennyi fontos információja megtalálható. Megkülönböztetünk normál és bővített FCB-t. Egy FCB lehet továbbá megnyitott és meg nem nyitott.

A normál file leíró blokk szerkezete:

Cím byte (decimálisan)	**Tartalom**
0.	Lemezegység azonosító, értéke: megnyitás előtt: 0 - aktuális egység 1 - A egység 2 - B egység stb. megnyitás után: 1 - A egység 2 - B egység stb. Ha megnyitás előtt az értéke 0, akkor megnyitó (0F) DOS funkció az aktuális egységnek megfelelően átállítja. Ha tartalmának megfelelő sorszámú egység nincs vagy hibás, akkor a funkció végrehajtása sikertelen lesz, és hibajelzést kapunk.
1-8.	File-név (kiterjesztés nélkül). A mezőben a file-nevet balra igazítva kell elhelyezni, és a 8 karakternél rövidebb nevek végét space-szel kell feltölteni.
9-11.	Kiterjesztés. A file-név kiterjesztését kell megadni benne, pont nélkül.
12-13.	Aktuális blokk sorszáma. Azt mutatja, hogy a file kezdetéhez viszonyítva hányadik 128 byte-ból álló blokk feldolgozása következik. A rekord méretét a logikai rekordhossz mező határozza meg. Az aktuális blokk sorszáma mezőt a file megnyitása (0F funkció) 0-ra állítja és a szekvenciális írás és olvasás funkcióhívások (14-es és 15-ös funkció) használják. (Egy blokk 128 db rekordból áll. Ezek hosszát a következő mező mutatja.)
14-15.	Logikai rekordméret (szó). Byte-okban megadja egy logikai rekord hosszát. A file megnyitásakor a rendszer az értékét 128-ra, azaz hexadecimális 80-ra állítja. Amennyiben ettől eltérő rekordmérettel akarunk dolgozni, úgy a megnyitás után át kell írni.

16-19. File-méret (duplaszó). A file byte-okban mért teljes hosszát tartalmazza. A méret a duplaszó szervezésnek megfelelően fordított sorrendben tartalmazza a számot.

20-21. Dátum (szó). A file létrehozásának, illetőleg utolsó módosításának dátumát tartalmazza év/hó/nap formában a következő módon:

```
|   21. byte    |    20. byte   |
 F E D C B A 9 8 7 6 5 4 3 2 1 0  bitek
|      év       | hónap |   nap |
```

22-31. Foglalt a rendszer számára.

32. Aktuális rekord sorszáma a blokkban. Értéke 0-127 között lehet, és megadja, hogy az aktuális blokkon belül hányadik rekord kerül feldolgozásra. Ezt a mezőt a 0F funkció nem inicializálja, ezért beállításáról a felhasználónak kell gondoskodnia.

33-36. Rekord abszolút sorszáma (duplaszó). A file elejétől számítva adja meg az aktuális rekord sorszámát. Az első rekord sorszáma 0. A sorszám értékét a random írás és random olvasás (lásd 28-as és 27-es funkció leírását) használja, és beállításáról a felhasználónak kell gondoskodnia.

A nem megnyitott FCB-ben csak a 0-11 byte-okat kell kitölteni. A további mezőket a file megnyitási, illetőleg -létrehozási funkcióhívások során a rendszer tölti ki.

Meg nem nyitott FCB-re találhatunk példát a PSP felépítését ismertető részben. Megjegyezzük, hogy ha a PSP-ben 5C címen kezdődő, az operációs rendszer által létrehozott FCB-t használjuk, figyelni kell arra, hogy annak utolsó byte-ja a 80-as címre esik. A 80-as címtől kezdődik a PSP-ben a DTA terület, ezért, ha ezt nem állítjuk át, az FCB vége átíródhat.

A normál FCB segítségével csak a normál attributumú file-ok kezelhetők. Ha a felhasználó rejtett, rendszer vagy csak olvasható file-okat akar kezelni, akkor bővített FCB-t kell használnia.

A bővített FCB egy, az elején egy 7 byte-os prefixet tartalmazó file leíró blokk.

A bővített file-leíró blokk szerkezete:

Cím byte (dec.) Tartalom

0. Jelző, értéke FF kell legyen. Ez jelzi a rendszer számára, hogy bővített FCB következik.

1-5. A rendszer számára foglalt.

6. File attributum. Azok a DOS funkciók, amelyek hívásakor FCB-t kell megadni, a megadott FCB-ben kijelölt attributumú (tulajdonságú, pl. rejtett) file-okkal fognak foglalkozni.

Az attributum byte bitjeinek jelentése:

6-7. bit értéke 0.

5. bit Archív bit. File-ba írás után értéke 0-ra vált, és a sikeres file-lezárás (10-es funkció) állítja 1-re (hexadecimális 20 a byte értéke normál és jól lezárt file-nál).

4. bit Altartalomjegyzék jelző. Ez a bejegyzés altartalomjegyzék nevet jelöl ki.

3. bit Kötetnév jelző. Ez a bejegyzés az adathordozó nevét jelöli. A byte többi bitje 0 kell legyen, azaz az attributum byte értéke ekkor 8.

2. bit Rendszer-file. A file csak speciális módon érhető el. Normál DOS műveletekben nem vesz részt.

1. bit Nem látható (rejtett, hidden) file. A file a DIR listán nem jelenik meg. Ezt az attributumot a file létrehozásakor, kreálásakor kell megadni.

0. bit Csak olvasható file. A file a PC DOS-szal közvetlenül nem törölhető, nem írható felül.
A 0-2. bitek tetszőleges értékkombinációja előfordulhat. Leggyakoribbak:
20 - normál file,
27 - csak olvasható, rejtett rendszer file,
21 - csak olvasható file.

7-41 Normál FCB. Az itt elhelyezkedő FCB teljes mértékben megegyezik az előzőekben ismertetekkel.

15.4.2 Lemezátviteli terület (DTA)

A lemezátviteli terület (Disk Transfer Area = DTA) az FCB-orientált file-kezeléskor az I/O puffer szerepét játsza, azaz a memóriának az a része, ahova az olvasási műveletek az adatokat a file-ból beírják, és ahonnan az írási műveletek az adatokat veszik.

A DTA méretét mindig a file leíró blokk decimális 14-15 byte-ján található érték mutatja. Ha ezt az értéket nem írjuk felül, akkor a DTA mérete 128, azaz hexadecimális 80 byte. (Ezek a file-műveletek mindig egy logikai blokkot mozgatnak a DTA-ról, illetőleg a DTA-ra.)

A DTA helyét a felhasználó az 1A funkció hívásával állíthatja be. Amennyiben nem állítjuk be, úgy a rendszer a DTA-t a PSP 80. byte-jától kezdi.

15.4.3 File-kezelés az FCB segítségével

Az FCB-t felhasználó DOS funkcióhívások a következő műveleteket végzik el:

file-műveletek:
- file megnyitása
- file létrehozása
- file lezárása

- szekvenciális (soros) írás
- szekvenciális (soros) olvasás
- közvetlen elérésű (random) írás
- közvetlen elérésű (random) olvasás

tartalomjegyzék kezelés:

- első bejegyzés keresése
- további, a megtalálttal egyező bejegyzés keresése
- file törlése
- file átnevezése

Ehhez a műveletcsoporthoz tartozó kiegészítő funkció a DTA beállítása.

A fenti funkciókat részletesen a DOS funkciókat leíró fejezetben ismertetjük majd.

Mint a fejezet bevezető részében leírtuk, ennél a file-kezelési módnál a file neve, a file-t tartalmazó adathordozó helye az FCB megfelelő mezőjében előzetesen beállítandó. Vagyis első lépésként mindig létre kell hozni egy megfelelő FCB-t.

Ezután kiadható egy file megnyitó vagy file létrehozó funkcióhívás. (Csak már létező file-t lehet megnyitni, de output esetén újabb rekordokkal kiegészíthető az így megnyitott file. Ezzel szemben tetszőleges nevű file-t létrehozhatunk, de ha létezett már ilyen nevű file, akkor annak korábbi tartalma elvész.)

A file-ba való írás, illetőleg a file-ból való olvasás előtt be kell állítani a DTA címét, vagyis a memóriában lévő munkaterületünk címét.

Közvetlen írási vagy olvasási műveleteknél be kell még állítani (ha szükséges) az elérni kívánt rekord vagy blokk mutatóját.

További kötöttsége ennek a file-kezelésnek, hogy a DTA mérete maximum 64 Kbyte lehet, de semmiképp sem lépheti túl a kijelölt szegmens felső határát. Ez utóbbi megkötés azt jelenti, hogy például 2000:F000 DTA kijelölése esetén a DTA maximum 2000:FFFF-ig terjedhet, azaz maximum hexadecimális 1000 byte-os blokk olvasható vagy írható.

Ezzel a funkciócsoporttal egy file szekvenciális beolvasásának menete a következő:

1. Kialakítunk egy meg nem nyitott FCB-t. Kezdődjön ez az INPUTFCB nevű tárterületen, és tartalmazza a "PROBA.KIT" nevet, továbbá jelezzük, hogy az aktuális egységről akarunk beolvasni. Ezt Microsoft MASM makró-assembler számára a következőképpen írhatjuk le. (A továbbiakban is a példákat így, MASM-ban adjuk meg.)

```
INPUTFCB DB 0                    ; egységazonosító
         DB "PROBA   "           ; file-név
         DB "KIT"                ; kiterjesztés
         DB 24 DUP(0)            ; 24 byte FCB részére
```

Legyen ez a terület a kódszegmensen elhelyezve. Ekkor a

```
PUSH CS
POP  DS
```

utasításpárral beállíthatjuk a megfelelő adatszegmens értéket. A továbbiakban feltételezzük, hogy az adatszegmens értéke mindig jól be van állítva.

2. A file megnyitása inputra, mely a következő utasításokkal érhető el:

```
MOV DX,OFFSET INPUTFCB      ; DS:DX beállítása FCB címére
MOV AH,0FH                  ; a funkció száma
INT 21H
                            ; az AH-ban megadott funkció
                            ; hívása (a DOS funkciók a
                            ; 21-es megszakításon keresztül
                            ; érhetők el).
```

Igényes program a funkcióhívás után ellenőrzi, hogy az sikeres volt-e. Itt most ettől eltekintünk.

```
MOV CX,0                    ; file méret
```

3. DTA beállítása. Használjuk erre a célra a program legvégén elhelyezkedő "MUNKA" nevű területet. A DTA beállításához szükséges utasítások:

```
MOV DX,OFFSET MUNKA         ; DS:DX beállítása
ADD DX,CX
MOV AH,1AH                  ; funkciószám
INT 21H                     ; funkcióhívás
```

4. 1 rekord beolvasása, feltételezve, hogy rekordonként akarunk olvasni.

```
MOV DX,OFFSET INPUTFCB      ; DS:DX beállítása FCB címére
MOV AX,14H                  ; funkciószám
INT 21H                     ; funkcióhívás
```

5. File vége vizsgálat

```
CMP AL,0                    ; összehasonlítás
JZ VAN_MEG                  ; nincs file vége, ennek
                            ; megfelelő címkére lép
CMP AL,1                    ; összehasonlítás
JZ VEGE1                    ; vége, elugrás
CMP AL,3                    ; hasonlítás
JZ VEGE2                    ; vége, elugrás
...
```

A VEGE1 és VEGE2 abban különbözik, hogy a VEGE2 esetén töredék blokk került beolvasásra és ennek méretét hozzá kell adni az eddig számolt mérethez, amelyet jelenleg a CX regiszter tartalmaz.

6. Ha nem volt file vége, akkor a DTA-t, továbbá a file méretét növelni kell.

```
ADD CX,128                  ; file-méret növelés
```

A 3-6. lépést a file végéig kell ismételni.

7. File lezárása.

```
MOV DX,OFFSET INPUTFCB      ; DS:DX beállítása az FCB címére
MOV AH,10H                  ; funkciószám
INT 21H                     ; CLOSE funkcióhívás
```

Megjegyezzük, hogy ez a fajta - 128 byte-os rekordokban való - olvasás lassú. Ezért nagyobb file-okat célszerűbb blokkonként írni/olvasni. Ezt az eljárást egy másik példán mutatjuk be.

Írjuk ki az előző példában beolvasott file-t az "A" egységre, "UJ"-ra módosított kiterjesztésű névvel. Ezt a feladatot a következő programmal valósíthatjuk meg:

1. lépés: FCB módosítása

```
MOV BYTE PTR [INPUTFCB],1       ; "A" egység
MOV BYTE PTR [INPUTFCB + 9],"U" ; kiterjesztés
MOV BYTE PTR [INPUTFCB + 10],"J" ; átírása
MOV BYTE PTR [INPUTFCB + 11]," " ; átírása
```

2. lépés: File létrehozása

```
MOV DX,OFFSET INPUTFCB      ; DS:DX beállítása FCB címére
MOV AH,16H                  ; file létrehozás
INT 21H                     ; funkcióhívás
```

3. lépés: Megnyitott FCB beállítása. Az FCB 32. byte-ját 0-ra kell állítani.

```
MOV BYTE PTR [INPUTFCB + 32],0
```

4. lépés: DTA beállítása

```
MOV DX,OFFSET MUNKA         ; DS:DX beállítása
MOV AH,1AH                  ; funkciószám
INT 21H                     ; funkcióhívás
```

5. lépés: Kiírandó blokkok számának kiszámítása.

```
PUSH CX                     ; file-méret mentése CX-be
MOV AX,CX
MOV CL,128                  ; blokkméret
DIV CL
MOV CL,AL                   ; eredmény CX-be
MOV CH,0
INC CX                      ; töredék blokk miatt
```

A 28-as funkcióhívás a CX-ben beállított darabszámú 128-byte-os blokkot ír ki folytonosan a DTA-ról, az FCB szerint.

6. lépés: Blokk közvetlen írása

```
MOV DX,OFFSET INPUTFCB      ; DS:DX beállítása
MOV AH,28H                  ; funkciószám
INT 21H                     ; funkcióhívás
```

7. lépés: File-méret beállítása az FCB-ben

Erre azért van szükség, mert a töredék blokk helyett teljes 128 byte-os blokkot írtunk ki. Ezt úgy is korrigálhatjuk, hogy a file lezárása előtt az FCB-be beírjuk a tényleges méretet.

```
POP CX                      ; méret visszahozása
AND CL,7FH
MOV [INPUTFCB + 16],CL      ; elég a méret végét átírni
```

8. lépés: File lezárása

```
MOV DX,OFFSET  INPUTFCB     ; DS:DX beállítása FCB címére
MOV AH,10H                  ; file-zárási funkció beállítása
INT 21H                     ; funkcióhívás
```

Látható, hogy több rekordot együtt írva lényegesen egyszerűbb a file kiírása, de kihasználtuk azt, hogy 64 Kbyte-nál kisebb file-ról van szó. Ennél nagyobb file esetén blokkos adatmozgatás használata esetén is ciklust kell szervezni.

Nem ilyen egyszerű a file olvasási-, illetőleg írási feladat, ha tetszőleges tartalomjegyzékben elhelyezkedő file-t kell mozgatni. Ekkor a file megnyitása előtt a 3B funkcióhívással ki kell jelölni a kívánt egységen a megfelelő tartalomjegyzéket aktuális tartalomjegyzéknek. Ha a file nevét például konzolról kérdezzük meg, akkor - az FCB felépítése miatt - kényelmetlen string átalakításra van szükség. Ugyanezek a megjegyzések igazak, ha nem írni vagy olvasni akarunk, hanem file-t keresünk, törlünk, vagy átnevezünk.

15.4.4 A DOS file-szám (handle) és használata

Láttuk, hogy az FCB-s file-kezelés elég sok kötöttséget jelent. Különösen érezhető ez, ha a különböző altartalomjegyzékekkel dolgozunk. Vigyázni kell arra is, hogy a feldolgozás kezdetére az FCB helyesen legyen kitöltve, és a feldolgozás során ne romoljon el. Lényegesen kényelmesebb munkát biztosít a file-számos file-kezelés. Ilyenkor a megnyitáshoz, törléshez elég a DOS megszokott formátumában megadni a file nevét. Anélkül, hogy az előző részhez hasonlóan itt is részleteznénk, nézzük végig itt is egy file beolvasásának vagy írásának vázlatos menetét.

1. File megnyitása. A megnyitáshoz ki kell alakítani egy karaktersorozatot, amely folyamatosan tartalmazza a file-t tartalmazó tartalomjegyzék elérési útját és a file nevét. A string végét egy nulla byte jelzi. (Ha csak file-nevet adunk meg, akkor az aktuális egység aktuális altartalomjegyzékét feltételezi a rendszer.)

Amennyiben file-névként a rendszer szabványos periférianeveit vagy a CONFIG.SYS-ben DEVICE által kijelölt periférianevet adunk meg, úgy a rendszer ezt a perifériát fogja fileként kezelni.

A megnyitáshoz a DS:DX mutatja a file-nevet tartalmazó string kezdetét, AL pedig a file elérési módot (0 = olvasás, 1 = írás, 2 = írás-olvasás) jelzi. A megnyitás a 3D funkcióhívással (INT 21, AH = 3D) történik. Sikeres megnyitás esetén a funkció AX-ben egy 4-nél nagyobb értéket ad vissza. Ez az érték a file-szám. A továbbiakban minden file-műveletnél a BX-nek ezt a számot kell tartalmaznia, mert ez mutatja a rendszernek, hogy melyik file-ra vonatkozik az utasítás. Ha az AX 0-4 közötti értéket tartalmaz, akkor a DOS standard input/output perifériáit akarjuk kezelni. (Ezeket a rendszer a program indítása előtt automatikusan megnyitja.) (Részletesebb információ a 3D funkcióhívás leírásánál található.)

2. Ha szekvenciálisan akarunk írni vagy olvasni, akkor itt következhet az író-, illetőleg olvasó utasítást tartalmazó ciklus.

Itt nincs megkülönböztetve szekvenciális, random rekord, valamint random blokk olvasás/írás.

Egyetlen író, illetőleg olvasó utasítás van. A mozgatandó byte-ok számát a CX regiszter tartalmazza. Az adatok memóriabeli kezdetét a DS:DX jelöli ki. (Itt nem okoz problémát, ha DX + CX > 64 Kbyte, azaz az adatmozgatás szegmenshatárt lép át, a rendszer lekezeli ezt az esetet is.)

A file-ban az adatmozgatás kezdetét egy, a file-hoz rendelt mutató, az úgynevezett file-pointer tartalmazza. A file-pointer egy 4 byte-os szám, amely tartalmazza, hogy a file elejétől számítva hányadik byte-tól akarunk írni, illetőleg olvasni.

A file megnyitás, illetőleg létrehozás a file-pointert 0-ra állítja, az írás és olvasás pedig a mozgatott terület végére.

A fentiekből következik, hogy szekvenciális file-kezelésnél a file-pointerrel nem kell foglalkozni.

3. Random file-kezelésnél szükség van a file-pointer kezelésére.

Erre a célra a 42-es funkcióhívás szolgál, amellyel a file kezdetétől, végétől vagy pedig aktuális értékétől számítva tetszőlegesen átállíthatjuk a file-pointert.

4. A munka végén 3E funkcióhívással le kell zárni a megnyitott file-okat. Ha ezt elmulasztjuk, output file-ok esetén a teljes file elvész.

Megjegyzés:

1. A 45-ös funkcióhívással mód van file-kezelő duplikálásra is. Ez azt jelenti, hogy ugyanarra a file-ra egyidejűleg két különböző file-számmal hivatkozhatunk.

2. Elsősorban nem igazi file-ok, hanem perifériákra mutató file-számok esetén van jelentősége, hogy a 44-es funkcióhívással lekérdezhetjük, illetőleg beállíthatjuk a file-ra vonatkozó vezérlő információkat. Például a funkcióhívással ellenőrízheti a program, hogy a feltételezett felhasználói perifériakezelő ténylegesen installálva van-e a rendszerben, vagy helyette egy valódi lemez-file-t nyitott-e meg a file-megnyitási utasítás.

16. DOS megszakítások

A DOS lehetőséget ad arra, hogy felhasználója hozzáférhessen a software megszakításokhoz (software interrupt-okhoz). Ezek használatán keresztül a rendszerprogramozó jobban megismerheti az operációs rendszer belső működését is. A fejezet tartalmának megértéséhez szükséges alapvető ismeretek (8086 processzor regiszterei, megszakítási rendszere, interrupt vektortábla felépítése, a példákban használt írásmód) megtalálható a DOS struktúráját ismertető korábbi fejezetben.

A PC DOS a megszakítások közül a hexadecimális 20-adiktól a 3F sorszámúig terjedőket használja. Az ezekhez tartozó megszakítási vektorok a 0. memóriaszegmens hexadecimális 80-as címtől az FF címig találhatók. A továbbiakban, ha nem jelezzük külön, akkor szám alatt (így a megszakítások és funkcióhívások sorszámai alatt is) hexadecimális számot értünk - bár ezt nem fogjuk mindig jelezni. Az egyes megszakításokat jelöljük majd így is:

INT *megszakítás sorszáma* pl.: INT 21

16.1 A 20-as megszakítás: program befejezése

Ezzel a megszakítással fejezhetik be a programok a futásukat és adhatják vissza a vezérlést a PC DOS-nak. Meghívásakor a rutin visszaállítja a felhasználói program indítása előtti Ctrl-Break-, kilépési- és kritikus hibákat kezelő vektor címet (INT 23, INT 22, INT 24). Valamennyi file-puffer tartalma törlődik. Használatakor, meghívása előtt, a következőkre kell figyelemmel lenni:

- minden, a felhasználói program által módosított file-t le kell zárni (lásd a 10., illetve 3E. funkcióhívást), különben a tartalomjegyzékben a file-méret, felírási idő, felírási dátum információk helytelenek lehetnek,
- gondoskodni kell róla, hogy a CS regiszter által kijelölt szegmens a Program szegmens prefix (PSP) vezérlőblokkra mutasson.

Mivel a DOS 4C és a 31 funkcióhívásaival a programból való visszatérés feladatát kényelmesebben meg lehet oldani - azokkal ugyanis visszatérési kódot is át tudunk adni -, célszerűbb azokat használni.

16.2 A 21-es megszakítás: DOS funkció meghívása

Részletes ismertetésére a 17. fejezetben kerül sor.

16.3 A 22-es megszakítás: program kilépési cím

A megszakítási vektortáblában a megszakításnak megfelelő helyen nem rutincím található, hanem az a cím, amelyre a DOS a program befejeződése után a vezérlést átadja. Új programszegmens létrehozásakor ezt a címet a DOS bemásolja a PSP-be. A megszakítás a felhasználók által nem hívható.

16.4 A 23-as megszakítás: Ctrl-Break kilépési cím

Az operációs rendszer ezt a megszakítást hívja meg, ha a <Ctrl-Break> leütése érvényre jut. Mint arról a "Rendszer konfigurálása" fejezetben szó volt, a <Ctrl-Break> billentyű felhasználó általi leütése csak akkor lesz hatásos, ha

BREAK=ON állapot esetén tetszőleges,
BREAK=OFF állapot esetén konzol, printer vagy aszinkron I/O műveletet megvalósító funkcióhívás kerül

végrehajtásra. (A 6-os és 7-es funkcióhívások végrehajtásakor lesz megszakítás.)

Amennyiben a felhasználó saját Ctrl-Break kilépési rutint ír és gondoskodik valamennyi regiszter lementéséről, úgy IRET (visszatérés interruptból) utasítással folytathatja a felhasználói programját.

A DOS Ctrl-Break megfelelő előkészítés után INT 20 segítségével visszalép a parancsfeldolgozóhoz. Ha a Ctrl-Break kilépési rutin használja a 9-es vagy az A-dik funkcióhívást, akkor a C-t a DOS kocsivissza, soremelés (CR,LF) karakterekre cseréli.

Ha IRET hatására a program tovább fut, akkor az aktuális I/O művelet új sor elején folytatódik.

A megszakítás érvényrejutásakor valamennyi regiszterben az az érték lesz, amelyet a regiszter az eredeti DOS funkcióhíváskor tartalmazott.

A felhasználói Ctrl-Break rutinra semmiféle megszorítás nincs, még a DOS funkciókat is szabadon használhatja. IRET kiadása előtt azonban valamennyi regiszterben a belépési értéknek kell lennie.

Ha a rutin a visszatérést távoli return (RET FAR) segítségével oldja meg, akkor a carry flag használható annak közlésére, hogy a program folytatódjék-e. (IRET esetén a CF értéke NO.)

Amennyiben egy program új programszegmenst nyit, és az ebben futó program kicseréli a Ctrl-Break megszakítás címét, úgy a második program befejezésekor, az első programba való visszalépés előtt, a DOS a második program PSP-jéből visszaírja az első program futása alatt érvényes Ctrl-Break címet.

A 22-es megszakításhoz hasonlóan, felhasználói programból ne hívjuk meg ezt az interruptot.

16.5 A 24-es megszakítás: kritikus hibák kezelése

A DOS akkor hívja meg ezt a megszakítást, ha kritikus hibát észlel.

Meghívásakor az AH regiszter tartalmazza a hiba típusát jellemző információkat. Az AH regiszter 7. bitje 0 lesz lemezhiba esetén. Karakteres berendezés vagy FAT hibájakor az AH további információkat csak lemezhiba esetén tartalmaz, ezért ezek ott kerülnek ismertetésre.

A BP:SI mutat az aktuális DHC blokk-ra. (Aktuális DHCB = Device Header Control Block = azon berendezés vezérlő információit tartalmazó blokk, amelynél a hiba fellépett.) A DHCB részletes ismertetése megtalálható "A PC DOS belső struktúrája" című 13. fejezet ezzel foglalkozó részében, továbbá a 44-es funkció leírásánál. Innen a hibára vonatkozó további információk nyerhetők.

A hibakód a DI regiszter alacsonyabb helyiértékű byte-jában található az alábbiak szerint:

Hibakód	Leírás
0	Olyan lemezre próbáltunk írni, amely írásvédett.
1	Ismeretlen egység.
2	A meghajtó nincs bekapcsolt állapotban.
3	Ismeretlen parancs.
4	Adathiba (CRC).
5	Hossz hiba.
6	Pozicionálási hiba.
7	Ismeretlen típusú adathordozó.
8	A szektor nem található.
9	A papír kifogyott a nyomtatóból.
A	Írási hiba.
B	Olvasási hiba.
C	Nem meghatározható hiba.

Mielőtt a PC DOS ezt a hibarutint meghívná, ötször próbálja meg a kijelölt műveletet elvégezni. Csak ha mind az öt kísérlet sikertelen volt, akkor kerül sor az INT 24 meghívására.

Az INT 24-be való belépés előtt az általános regisztereket a PC DOS úgy állítja be, hogy a hibás művelet azok alapján újra ismételhető legyen.

A felhasználó természetesen ezt a hibakezelő rutint is kiválthatja saját eljárással. Ahhoz, hogy a DOS hibakezelés a felhasználó hibakezelő rutinjához kerüljön, majd ezt követően a DOS-hoz térjen vissza, a felhasználónak, mielőtt a 24-es megszakítás bekövetkezhetne, el kell tennie a 24-es interrupt vektort, és ezt olyannal kell helyettesítenie, amely a felhasználói hibakezelő rutinra mutat.

A 24-es megszakítás bekövetkezése után, amikor rutinunk megkapta a vezérlést, a flag regisztert a verembe kell tenni (PUSHF), és meg kell hívni az eltett INT 24 interrupt vektort (CALL FAR). Ekkor a DOS hibaüzenetet ad ki és vár, hogy a felhasználó milyen választ ad (Abort, retry, ignore). Ezután a DOS visszaadja rutinunknak a vezérlést a CALL FAR utáni utasításra.

A felhasználói verem (stack), amely a következő felépítésű (a verem tetejétől számítva), felhasználható arra, hogy a hiba lekezelése után ne térjünk vissza a DOS-hoz.

FLAG-ek
CS
IP visszatérés a 24-es megszakításból a felhasználói programhoz
ES

DS
BP
DI
SI Felhasználói regiszterek az eredeti 21-es funkcióhívás előtt
DX
CX
BX
AX
FLAG-ek
CS Visszatérés a hibakezelésből abba a DOS funkcióba, ahol a hiba fellépett
IP

Ilyen közvetlen visszatérésnél a verem tetejétől számított decimális 13., 14., 15. byte-on található FLAG, CS, IP-re kell az IRET utasítást kiadni (pl. úgy, hogy az IRET előtt D-vel megnöveljük az SP értékét). Arról a felhasználónak kell gondoskodnia, hogy a visszatérés utáni első funkcióhívásnak 12-nél magasabbnak kell lennie, különben a PC DOS meghatározatlan állapotba kerül.

Az INT 24-et lezáró IRET utasítás végrehajtása után a DOS az AL regiszter tartalmának megfelelően a következőképpen jár el:

(AL) = 0 A hiba figyelmen kívül marad.
= 1 A művelet ismétlése.
= 2 A program a 23-as megszakítással befejeződik.
= 2 Rendszerhívási hiba.

A felhasználó által írt hibakezelő rutinnak a következő regisztereket mentenie kell: SS, SP, DS, ES, BX, CX, DX, továbbá csak a 12-nél alacsonyabb rendszerfunkciókat használhatja. Ügyelni kell arra a felhasználó által írt hibakezelő rutin írásakor, hogy az INT 24-et a DOS mindig interrupt - letiltott - állapotban adja ki.

Lemezhiba

Lemezhibák esetén ez a rutin csak akkor aktiválódik, ha a hiba DOS funkció végrehajtása során fordult elő. (Pl. INT 21, AH = 14, szekvenciális olvasás.) Nem kerül meghívásra az INT 24, ha a lemezhiba INT 25 vagy INT 26 során lép fel.

Ha lemezhiba fordul elő (az AH 7. bit-je 0), akkor az AL regiszter a hibás meghajtó azonosítóját tartalmazza. (A bitek az alacsonyabb helyiértéktől számítva: 0 = A:meghajtó; 1 = B:meghajtó stb.) Az AH 0-2 bitjei pedig a következő információt tartalmazzák:

0. bit = 0 a hiba olvasási műveletnél
= 1 a hiba írási műveletnél történt
2.- 1. bit a lemez terület típusára utal:
0 0 DOS terület - rendszer állományok
0 1 FAT terület
1 0 tartalomjegyzék
1 1 adatterület

Ne feledjük, hogy a hibakód ilyen esetben is a DI regiszter alacsony helyiértékű byte-ján található. (Lásd az INT 24-ről általánosan leírtakat.)

Az AH regiszter 3-5. bitjei mutatják, hogy a hiba kezelésére milyen válasz adható.

Bit	Jelentés
3	=0 FAIL nem megengedett
	=1 FAIL megengedett
4	=0 RETRY nem megengedett
	=1 RETRY megengedett
5	=0 IGNORE nem megengedett
	=1 IGNORE megengedett

Ilyenkor értelemszerűen a BP:SI nem tartalmaz használható információt.

Egyéb hibák

Ha az AX regiszter 7. bitje 1, akkor a hiba karakteres berendezésen lépett fel, vagy az is előfordulhat, hogy a FAT bejegyzés hibás. Ekkor a DHC blokkból eldönthető, hogy melyik eset fordult elő: Ha az attributum byte magas helyiértékű bitjei blokk-orientált egységre utalnak, akkor a FAT hibás. Különben a hiba karakteres berendezésen fordult elő.

Ha karakteres berendezésről van szó, akkor az AL tartalma semmitmondó. Mint az INT 24 elején írtuk, a BP:SI mutat arra a DHCB-re, amelynél a hiba felmerült. Ez a következőket tartalmazza:

Cím (hányadik típus byte)	Adat	Az adat értelmezése
0-3.	duplaszó	A következő DHC blokk kezdő címe. Ha értéke FFFF, akkor ez az utolsó létező karakteres berendezés DHCB-je.
4-5.	szó	A berendezés típusának leírása, az alábbi bit kiosztás szerint: 15. bit = 0, akkor blokkos berendezés = 1, akkor karakteres berendezés 14. bit tartalmazza az Input-Output Controll (IOCTL) bitet 3. bit = 1, ha az aktuális CLOCK- (óra-) egységről van szó 2. bit = 1, ha az aktuális NUL-egységről van szó 1. bit = 1, ha az aktuális standard output egységről van szó 0. bit = 1, ha az aktuális standard input egységről van szó. (Értelemszerűen a 0.-3. bitek közül maximum az egyik lehet egyenlő eggyel. A fel nem sorolt bitek tartalma nulla.)
6-7.	szó	A handler által feldolgozandó parancs-blokk értelmezését végző rutin címe.
8-9.	szó	Az aktuális berendezés parancsfeldolgozó rutinjának címe.
A-11.	8 byte	Név mező. A karakteres berendezés neve. (Blokkos berendezés esetén a handler által kezelt egységek számát tartalmazza.)

Tehát ha a BP:SI + 4. címen lévő szó 15. bit-je 0, azaz blokkos berendezésről van szó, akkor a 4. szónak csak a 13. és 14. bit-je lehet nullától eltérő. A 13. bit abban az esetben egy, ha a lemez nem IBM DOS formátumú.

Karakteres berendezés esetén a BP:SI + A. címtől kezdődő 8 byte-ról tudjuk megállapítani a berendezés nevét.

A DHC részletes ismertetése a felhasználó által írt perifériakezelőkkel foglalkozó fejezetben található.

16.6 A 25-ös megszakítás: közvetlen lemezolvasás

A vezérlés közvetlenül a DOS BIOS-ra kerül. Visszatéréskor az eredeti flag-ek a veremben vannak, mivel azokat az INT utasítás lementette. Ezért visszatérés után a program jellemzőket vissza kell állítani. Az interrupt hívás bemenő paraméterei:

AL	A meghajtó azonosítója. (0 = A, 1 = B, 2 = C stb.)
CX	Az olvasandó szektorok száma.
DX	Az első szektor logikai száma.
DS:BX	Átviteli terület cím. (DTA)

A logikai szektorszám a szektorok következőképpen értelmezett egymásutáni elhelyezkedéséből adódik: a 0-ás sávon, a 0-s fejjel elérhető, fizikailag 1-es szektor a logikai 0. szektor, 0. fejjel elérhető fizikailag 2. a logikai 1. stb., majd a 0. fej utolsó (8. vagy 9.) szektorát kétoldalas lemeznél az egyes fej (lemezoldal) 0. sávjának első szektora követi, és így folytatódik a számozás az utolsó sáv utolsó fejének utolsó szektoráig. Ezért, bár a szektorok növekvő sorrendben vannak számozva, fizikailag nem biztos, hogy egymás mellett állnak. A 25-ös megszakítás meghívása esetén a szegmens regiszterek kivételével valamennyi regiszter tartalma törlődik. Ha az átvitel sikeres volt, akkor a CF = 0 (Carry Flag), ellenkező esetben CF = 1.

Hiba esetén az AX a hibára mutat. Az AL a DOS-hibakódját tartalmazza, amely azonos azzal, amelyet a DI alacsonyabb értékű byte-jában akkor kapunk, ha INT 24-et hív a DOS.

Az AH regiszter pedig hiba esetén a következő értékeket tartalmazhatja (mint mindig, hexadecimálisan értve):

80	A meghajtó üres, nincs benne floppy.
40	A pozicionálás hibás.
20	Vezérlőegység hiba.
10	Hibás CRC a lemez olvasásnál.
08	DTA-túllépés a műveletnél.
04	A kért szektor nem található.
03	Írásvédett lemezre próbáltunk írni.
02	Egyéb hiba.

16.7 A 26-os megszakítás: fizikai szintű lemezírás

Ami a 25-ös megszakításra vonatkozott, az itt is érvényes, csak a műveletnél olvasás helyett mindenütt írás értendő.

16.8 A 27-es megszakítás: befejezés, rezidens programok részére

Ezt azok a programok használják, amelyek különböző okok miatt lefutásuk után is a memóriában maradnak. Ehelyett jobb a 31-es DOS funkciót használni, mert azzal 64 Kbyte-nál nagyobb program is a memóriában maradhat, és RETURN (ERROR CODE vagy COMPLETION CODE) információk is átadhatók a DOS számára.

A 27-es interrupt meghívása előtt a programnak a DX-be kell töltenie az ő utolsó címe utáni cím értékét, ez offset érték, DS-hez viszonyítva, és DS = ES kell legyen. A DOS a 27-es megszakítással befejezett programot úgy kezeli, mint a DOS egy kiterjesztését, így ez a program akkor sem íródik felül, ha más programokat töltünk be és hajtunk végre. Így például felhasználó által írt megszakítás-kezelő programok esetén, amelyeknek állandóan a memóriában kell lenni.

28-2E megszakítások: a DOS számára fenntartottak

16.9 2F-es megszakítás: multiplex megszakítás

A 2F megszakítás a DOS olyan külső parancsaival való kapcsolattartásra szolgál, amelyek az első meghívásuk után a memóriában maradnak. (PRINT, ASSIGN, SHARE, APPEND). Ezen túlmenően a felhasználó szabadon definiálhatja a megszakítás használatát.

Minden egyes funkcióhoz tartozik egy ún. multiplexer szám. Az egyes funkciókat láncba kell fűzni, azaz amikor saját multiplexer programot írunk, az az inicializáló részben gondoskodik a 2F megszakítási vektor átírásáról. A rezidens rész, ha megkapja a vezérlést, ellenőrzi, hogy neki szóló multiplexer számot kapott-e. Ha nem, akkor átadja a vezérlést a lánc következő tagjának. Ha igen, akkor elvégzi a megfelelő feladatot.

A DOS-ba beírt funkciók úgy vannak elkészítve, hogy az AH-ba mindig a multiplexer számot míg az AL-be az elvégzendő feladat kódját kell beírni. A jelenleg érvényes párosítások az alábbiak:

PRINT rezidens része (AH = 1)

A multiplexer program a következő feladatokat tudja elvégezni:

AL = 0 *Installálási állapot lekérdezése.*

Visszatéréskor az AL arra vonatkozó információt tartalmaz, hogy a PRINT rezidens részét installáltuk-e már:

- AL = 0 még nem, de lehetséges
- AL = 1 még nem, és nem is lehet
- AL = FF installált.

AL = 1 *File leírása a nyomtatási sorba*

DS:DX a megszakítás meghívása előtt egy byte-ból és egy mutatóból álló memóriarészre mutat. A byte a verziót írja le (DOS 3.10-3.30 esetén kötelezően 0). A duplaszó eltolás, szegmens sorrendben a file-t definiáló ASCIIZ string elejére mutat. A file-név nem tartalmazhat * és ? karaktereket.

AL = 2 *File(ok) törlése a nyomtatási sorból*

DS:DX a sorból törlendő file-név elejére mutat. Az ASCIIZ string tartalmazhat * és ? karaktereket is.

AL = 3 *A nyomtatási sor törlése*

A funkció valamennyi file-t törli a nyomtatási sorból.

AL = 4 *Állapot lekérdezés kezdete*

Visszatéréskor DX a hibák számát tartalmazza. Ez a PRINT sikertelen kísérleteinek számát mutatja (miközben az utolsó karaktert nyomtatja). Ha DX = 0, akkor nem volt hiba. A funkció felfüggeszti a PRINT működését. A DS:SI a nyomtatási sorra mutat. A sor pontosan 64 byte hosszú bejegyzéseket tartalmaz, amelyek a nyomtatandó file-neveket tartalmazzák. Az első bejegyzés az éppen nyomtatott file neve. A sor végét egy olyan bejegyzés jelzi, melynek első karaktere 0.

AL = 5 *Állapot lekérdezés után továbbindítás*

A hívás hatására a sor nyomtatása - amit az AL = 4 hívás felfüggesztett - folytatódik. Visszatéréskor AX a hiba kódját tartalmazza. Az egyes kódok jelentése a következő:

AX	Jelentés
1	érvénytelen funkció
2	a file nem létezik
3	az út nem létezik
4	túl sok nyitott file
5	az elérés tilos
8	a sor megtelt
9	az eszköz foglalt
12	a név túl hosszú

AL = F8-FF: *a DOS számára fenntartott*

ASSIGN rezidens része (AH = 2)
SHARE rezidens része (AH = 10)
APPEND rezidens része (AH = 67)

Az AL = 0 esetben a rezidens rész installálásáról kapunk információt, úgy, mint a PRINT esetén. Egyéb funkció nincs.

17. DOS funkciók

A DOS a perifériák (lemezegységek és az összefoglaló néven karakteres berendezéseknek nevezett klaviatúra, display és nyomtató) I/O műveleteit, file-ok kezelését, a memória kezelését, dátum- és időbeállítását és egyéb műveleteket ellátó programokhoz, az úgynevezett DOS funkciókhoz biztosítja a felhasználó hozzáférését. Az alábbiakban megadjuk e funkciók hívási számát (hexadecimálisan) és tudnivalóit. A DOS funkciók csoportjai:

I/O műveletek karakteres berendezésekkel

01 = tasztatúra input
02 = kiírás képernyőre
03 = soros vonali input
04 = soros vonali output
05 = karakter kiírása nyomtatóra
06 = közvetlen konzol I/O
07 = közvetlen konzol input, képernyőre írás nélkül
08 = konzol input, képernyőre írás nélkül
09 = karaktersorozat kiírása
0A = pufferelt tasztatúra input
0B = standard input egység állapotának ellenőrzése
0C = a standard input puffer törlése, és egy standard input kezelő rutin hívása

Processzor-kezelő funkciók

00 = program befejezése (helyette inkább a 4C-t használjuk!)
26 = új program szegmens előállítása (helyette inkább 4B-t használjuk!)
31 = programfutás befejezése úgy, hogy a program a memóriában marad
4B00 = program betöltése és végrehajtása (EXEC)
4B03 = program betöltése végrehajtás nélkül
4C = program befejezése
4D = a hívott program befejezési kódjának lekérdezése (WAIT)
62 = program szegmens lekérdezése

Memória kezelés

48 = memória foglalás
49 = lefoglalt memória felszabadítása
4A = lefoglalt memóriablokk méretének megváltoztatása

Tartalomjegyzékkel kapcsolatos műveletek

39 = altartalomjegyzék létrehozása (MKDIR)
3A = altartalomjegyzék törlése (RMDIR)
3B = aktuális altartalomjegyzék beállítása (CHDIR)
41 = file törlése (UNLINK)

43 = file attributumának lekérdezése vagy beállítása
47 = adott meghajtó aktuális altartalomjegyzékének lekérdezése
4E = elérési úttal megadott file keresése az első találatig
4F = további, a megadottnak megfelelő file keresése (4E után)
56 = file átnevezése
57 = file dátumának és időpontjának lekérdezése, illetve módosítása

FCB-s file-kezelő funkciók (használatuk ellenjavalt!)

0F = file megnyitása
10 = file lezárása
11 = adott FCB-beli file-név keresése a tartalomjegyzékben
12 = a 11-es funkcióval megtalált file-t követő további egyező file keresése
13 = file törlése
14 = file soros olvasása
15 = file soros írása
16 = file létrehozása
17 = file átnevezése
21 = random olvasás
22 = random írás
23 = file-méret lekérdezése
24 = relatív rekord mező beállítása
27 = random blokk olvasása
28 = random blokk írása

File-számos file-kezelés

3C = file létrehozása, vagy meglevő file hosszának 0-ra állítása
3D = file megnyitása
3E = file lezárása
3F = olvasás file-ból vagy egyéb perifériáról
40 = irás file-ba vagy egyéb perifériára
42 = írási/olvasási mutató állítása
45 = file file-számának duplikálása
46 = file-szám duplikálása úgy, hogy a másodpéldányt a felhasználó adja meg
5A = file létrehozása egyedi névvel
5B = új file létrehozása
67 = file-számos periféria-kezelők maximális számának megadása
68 = puffer kiürítése

Egyéb rendszerfunkciók

0D = lemez alaphelyzetbe állítása
0E = lemezegység kiválasztása
19 = az akuális meghajtó kódjának bekérése
1A = a DTA címének beállítása
1B = az aktuális meghajtó információinak bekérése
1C = adott meghajtó információinak bekérése

25 = a megszakítási (interrupt) vektor beállítása
29 = file-azonosító keresése adott parancssorban
2A = dátum lekérdezése
2B = dátum beállítása
2C = rendszeridő lekérdezése
2D = rendszeridő beállítása
2E = VERIFY (lemezírás utáni ellenőrzés) kapcsoló beállítása
2F = lemez átviteli terület (DTA) címének lekérdezése
30 = DOS verziószám lekérdezése
33 = Ctrl-Break lekérdezése és beállítása
35 = megszakítás (interrupt) vektor lekérdezése
36 = szabad lemezterület lekérdezése
38 = ország-specifikus információk lekérdezése
54 = VERIFY (lemezírás utáni ellenőrzés) kapcsoló lekérdezése
65 = kibővített ország-specifikus információk lekérdezése
66 = globális kódlap beállítása/lekérdezése

File megosztás és hálózatkezelés

3D = file megnyitása adott elérési móddal
4409 = átirányítás ellenőrzése blokkos berendezésre
440A = átirányítás ellenőrzése karakteres berendezésre
440B = elérési kísérletek számának beállítása
5C00 = file zárolása
5C01 = file zárolásának feloldása
5E00 = lokális név lekérdezése
5E02 = nyomtató inicializálása
5F02 = hozzárendelés lekérdezése
5F03 = hozzárendelés beállítása
5F04 = hozzárendelés törlése

A 2F-57 funkciók a DOS 2.00-ban jelennek meg. Azokban az esetekben, amikor hasonló funkciók kisebb és nagyobb hívószámmal is szerepelnek, akkor az újak (nagyobb számúak) használatát javasoljuk. Ezek definíciója egyszerűbb és így teljesítményük jobb, mint a hagyományos párjainak. Ha a programban az új funkcióhívásokat használjuk, akkor azokat a DOS 1.00 vagy 1.10 operációs rendszerben nem lehet végrehajtani.

A DOS 3.00-ás verziótól lehet az 55-5F funkciókat használni, amelyek a hálózatkezeléssel foglalkoznak.

Ha a DOS a vezérlést átveszi, akkor egy belső veremre kapcsol át. A felhasználói regisztereket ebbe menti, azért, hogy az igényelt függvényhívás az információkat vissza tudja adni.

A felhasználói területen lévő veremnek elég nagynak kell lennie ahhoz, hogy a megszakítási rendszer az összes regisztert le tudja menteni.

Azt ajánljuk, a felhasználói veremben azon kívül, amit a felhasználó akar benne tárolni, legyen legalább 80 (hexadecimális) byte-nyi hely a DOS számára.

17.1 A DOS funkciók hibakezelése

Ha a művelet sikeresen befejeződött, akkor az előbb említett új funkcióhívások többsége a flag-et törölt átviteli bit-tel (Any Flag) adja vissza (No Carry), hiba esetén pedig beállítja az átvitelt (Carry Yes). Ez utóbbi esetben az AX a következő bináris hibakódokat tartalmazhatja:

Kód	Leírás
1	Érvénytelen függvényhívás
2	A file nem található
3	Az elérési út nem található
4	Túl sok megnyitott file
5	Az elérés nincs engedélyezve (hálózati alkalmazásoknál)
6	Érvénytelen kezelés
7	A memóriakezelő blokkok lánca megsérült
8	Nincs elég memória
9	Érvénytelen memória-blokk cím
10	Érvénytelen környezet
11	Érvénytelen forma
12	Érvénytelen elérési kód
13	Érvénytelen adatok
15	Érvénytelen meghajtó megadás
16	Kísérlet az aktuális könyvtár törlésére
17	Nem ugyanaz az egység
18	Nincs több file
19	Írásvédett lemez
20	Rossz lemezegység
21	A lemezegység nincs készenléti állapotban
22	Érvénytelen lemezműveleti parancs
23	CRC hiba
24	Nem megfelelő hossz
25	Keresési (seek) hiba
26	Nem DOS lemez
27	Szektort nem találja
28	Kifogyott a papir
29	Iráshiba
30	Olvasási hiba
31	Általános hiba
32	Megosztás nem megengedett
33	Zárolás nem megengedett
34	Rossz lemez
35	FCB nem elérhető
36	puffer túlcsordulás a megosztásnál
37-49	Fenntartott
50	Nincs hálózat
51	A távoli számítógép nem 'hallgató'

Kód	Leírás
52	Dupla név a hálózatban
53	A hálózati nevet nem találja
54	A hálózat foglalt
55	A hálózaton implementált eszköz nem létezik már
56	A hálózati BIOS határát túlléptük
57	Hálózati adapter hiba
58	Hibás válasz a hálózatból
59	Felismerhetetlen hálózati hiba
60	Nem kompatibilis hálózati adapter
61	A nyomtatási sor megtelt
62	A nyomtatási sor nincs tele
63	Nincs elég hely a nyomtatási file-hoz
64	A hálózati nevet törölték
65	Az elérés nincs megengedve
66	A hálózati eszköz típus nem megfelelő
67	A hálózat neve nem létezik
68	A hálózati név határát túlléptük
69	A hálózati BIOS szelet határát túlléptük
70	Ideiglenesen szünetel
71	A hálózati kérés nem szolgálható ki
72	A lemez vagy nyomtató átirányítása szünetel
73-79	Fenntartott
80	A file létezik
82	Nem hajtható végre
83	24-es típusú megszakítási hiba
84	A struktúrán kívül van
85	Már hozzárendelték
86	Érvénytelen jelszó
87	Hibás paraméter
88	Hálózati adathiba

Az 1-18 hibák a DOS 2.10-es változatától szerepelnek, s azonosak az INT 21 megszakítás által kezelt hibákkal. Hasonlóan a 19-31 sorszámú hibák azonosak az INT 24 megszakítás által kezelt 0-12 sorszámú hibákkal. A 32-83 hibákat a DOS 3.00-ás verzióitól kezdve jelenti a rendszer, s elsősorban a file-megosztással és a hálózatkezeléssel kapcsolatosak. A 84-88 hibakódok a 3.10-es verzióban jelennek meg először.

17.2 A DOS funkcióhívások általános szabályai

A kívánt funkció számát mindig az AH regiszterbe kell tenni. Ez a szám az a hexadecimális szám, amellyel az egyes funkciókat itt le fogjuk írni. Egyes funkciók más regiszter beállítását is megkívánják, ezeket, és az esetleges kimeneti regisztertartalmakat az illető funkciók leírásánál adjuk majd meg.

Egyes funkcióhívások inputja ASCII jelsorozat is lehet. Ebben az esetben az ASCII jelsorozat meghajtó-, elérési út- és néhány esetben file-névből áll. A jelsorozat egy bináris

0-t tartalmazó byte-tal fejeződik be. Figyelemmel kell lenni arra, hogy azok a funkcióhívások, amelyek útmegadást elfogadnak, elválasztóként a \-t (backslash) használják.

Azok az új funkcióhívások, amelyek a file-kezeléssel foglalkoznak, használnak egy numerikus azonosítót, amelyet "file-szám"-nak (handle) nevezünk. Ha ezekkel állítunk elő, vagy nyitunk meg file-t, vagy egységet, akkor az AX tartalma egy 16 bites bináris érték lesz, amelyet file-szám-ként kapunk vissza. Ez a file-szám azonosítja a megnyitott file-t. A DOS-ban a következő file-számok fenntartottak:

0000 Standard input egység (az input átirányítható).
0001 Standard output egység (az output átirányítható).
0002 Standard hibakiíró-egység (az output nem irányítható át).
0003 Standard vonali egység.
0004 Standard nyomtatóegység.

Ezeket megnyitás nélkül lehet a felhasználói programokban használni.

A legtöbb funkció hívásakor a paramétereket regiszterekben kell átadni. Az egyes funkcióknál ismertetjük majd, hogy mely regiszterek játszanak szerepet. Az esetleges paraméterbeállítás után a funkciók hívása a következő eljárások valamelyikével történik:

1. A funkciót azonosító értéket az AH-ban kell beállítani. A program szegmens prefixben (PSP) hexadecimális 50-re adjunk ki egy CALL FAR szubrutinhívást. (Itt figyelembe kell venni azt, hogy azok a programok, amelyek ezeket a funkcióhívásokat használják, a DOS 1.00 és az 1.10 operációs rendszer alatt nem működnek jól.)

2. A funkcióhívás azonosító értékét az AH-ban beállítjuk, majd kiadjuk a 21-es megszakítást (INT 21).

3. Léteznek programok, amelyeket más hívási konvenciókkal írtak meg. Ezek a funkció számát a CL-be teszik, a többi regisztert a funkcióhívás specifikációjának megfelelően állítják be. Ezután a PSPA + 5-re egy közeli CALL-t adnak ki. Ezen a címen egy CALL FAR van, amely a DOS funkciókat adminisztráló programot hív meg. Ennél a módszernél az AX-regiszter tartalma mindig törlődik. (CP/M konvenció. Használata ellenjavalt!)

Megjegyzés:

1. A funkcióhívásokra általánosan igaz:
   - A funkcióhívás kódját az AH regiszterben kell megadni;
   - A memóriaterületek címét a DS:DX regiszterekben kell megadni;
   - A funkcióhívás eredménye az AX regiszterbe kerül.
2. Az assembly, 'C' és a PASCAL nyelvből közvetlenül is meg lehet a funkciókat hívni. A funkcióhívásokat viszont közvetve minden magasszintű nyelvből elérhetjük, ha az adott programozási nyelvből meghívjuk a megfelelő assembler programot.
3. A FCB-vel foglalkozó funkcióhívások speciális karaktereket (0D-29 hexadecimális) nem engednek meg.

4. A funkcióhívások az AX regiszter tartalmát megváltoztathatják, így arra ne számítsunk, hogy ott változatlan érték marad.

5. A funkcióhívásokra vonatkozó példákat Turbo Pascal-ban adjuk meg. Ennek oka, hogy ez sokkal könnyebben olvasható, mintha pl. MACRO ASSEMBLER-ben adnánk meg. A példák megadásakor feltételezzük, hogy a Turbo Pascal program az alábbi deklarációkat tartalmazza:

```
type
  RegisterSet = record case integer of
                1: (AX,BX,CX,DX,BP,DI,SI,DS,ES,Flags: integer);
                2: (AL,AH,BL,BH,CL,CH,DL,DH: Byte);
                end {record};
   buffer       =array[1..32] of byte;
   szoveg       =string[255];
   mutato       =^buffer;
   integer2     =record
                    LowInt,HighInt: integer
                 end;
   fcbset        =record
                    Drive: byte;
                    Name: string[11];
                    Extention: string[3];
                    CurrBlock: integer;
                    RecodSize: integer;
                    FileSize: integer2;
                    Date: string[2];
                    Time: string[2];
                    Reserved: string[8]
                    CurrRecord: byte;
                    RelRecord: integer2
                 end;
var reg: RegisterSet;
```

17.3 A DOS funkciók ismertetése

17.3.1 0-s funkció - program befejezése

A funkció a programot úgy fejezi be, hogy az INT 22, 23 és 24 címeket visszaállítja azokra az értékekre, amelyeket a programozó a funkció hívása előtt a PSP-be kimentett.

A funkció valamennyi adatpuffert lezárja (kiüríti). Azok a file-ok, amelyeket módosítottunk, a régi jellemzőkkel kerülnek a tartalomjegyzékbe (dátum, hossz stb.). Ezért ezeket előbb le kell zárni (lásd a 10-es funkciót!). A vezérlés a befejezési címre kerül. Ez a funkcióhívás ugyanazt hajtja végre, mint az INT 20. A funkció hívása előtt gondoskodni kell arról, hogy a CS regiszterbe kerüljön a PSP szegmenscíme. Ez a funkció egyik regiszterben sem ad vissza információt. (A DOS-hoz kerül a vezérlés.)

17.3.2 1-es funkció - tasztatúra input

A standard input egységről karakter beírását várja, majd ezt a jelet a standard output egységen is visszaírja (echo). Visszatéréskor a karakterkód az AL regiszterbe kerül. Ha ez a karakter éppen a <Ctrl-Break>, akkor a 23-as megszakítás is végrehajtódik.

Megjegyzés:

Az 1-es, 6-os, 7-es és 8-as rendszerfunkciók esetén a kibővített ASCII kódok két funkcióhívást igényelnek. Az első hexadecimális 0-t ad vissza, amellyel jelzi a rendszer, hogy kiterjesztett kód következik.

Példa:

1.
```
function KbdEcho: char;
begin
   reg.AH:=$01;
   MSDos(reg);
   KbdEcho:=chr(reg.AL)
end;
```

2.
```
procedure ExtKbdEcho (varC: char,var E: boolean)
begin
 C:=KbdEcho;
 if C=#0 then
 begin
   C:=KbdEcho;
   E:=true
 end
 else E:=false
end;
```

A KbdEcho függvény azonos a Turbo Pascal 4.0-ás verziójában levő ReadKdb függvénnyel.

17.3.3 2-es funkció - kiírás képernyőre

A DL-ben lévő karakter a standard outputra kerül. Ha a karakter a <balranyíl>, akkor a kurzor egy hellyel balra lép és erre a helyre egy szóköz jel kerül.

A funkció a 09(TAB), 0A(LF) és 0D(CR) kódokat megfelelően értelmezi.

Példa:

```
procedure DisplayChr(ch:char);
begin
   reg.DL:=ord(ch);
```

```
   reg.AH:=$02;
   MSDos(reg)
end;
```

17.3.4 3-as funkció - soros vonali input

A standard soros vonalról vár egy karaktert. A jelet az AL regiszterben kapjuk meg.

Megjegyzés:

1. A soros vonalak (AUX, COM1, ..., COM4) pufferolás nélkül hajtják végre az I/O műveleteket.
2. Indításkor a DOS az első soros vonalat a következő kezdeti állapotra állítja be: 2400 baud, paritás nélkül, 1 stop-bit, 8 adat-bit.
3. A soros vonalak rendszerhívásai (3, 4) állapot- és hibakódot nem adnak. Az ellenőrzés érdekében ezért javasoljuk a ROM BIOS rutin (INT 14) használatát.

Példa:

```
function ReadAux: byte;
begin
   reg.AH:=$03;
   MSDos(reg);
   ReadAux:=reg.AL
end;
```

17.3.5 4-es funkció - soros vonali output

A DL-ben lévő jel a standard soros vonalra kerül. Ez a funkció egyik regiszterben sem ad vissza információt.

Példa:

```
procedure WriteAux(ch: byte);
begin
   reg.DL:=ch;
   reg.AH:=$04;
   MSDos(reg)
end;
```

17.3.6 5-ös funkció - karakter kiírása nyomtatóra

A DL-ben lévő karakter a standard nyomtatóra kerül. Ez a funkció egyik regiszterben sem ad vissza információt.

Példa:

```
procedure PrintChar (ch:char);
begin
   reg.DL:=ord(ch);
   reg.AH:=$05
   MSDos(reg)
end;
```

17.3.7 6-os funkció - közvetlen konzol I/O

Az input és az output között a DL regiszter tartalmának beállításával választhatunk.

- FF - input
- bármi más - output

Konzol input funkció végrehajtása után, ha van az input pufferban várakozó karakter, a Zero Flag törlődik (clear) és AL-ben megkapjuk a beolvasott karaktert. Ha nincs várakozó karakter, a Zero Flag értéke 1 lesz (set).

Konzol output funkció hívása előtt DL-be be kell tenni a standard outputra kiírandó karaktert.

A funkció nem figyeli sem a <Ctrl-Break>-et, sem a <Ctrl-PrtSc>-t.

Kibővített ASCII kódú billentyű lenyomásának értékelésére két funkcióhívásra van szükség. Az első esetben #0 karaktert ad vissza a funkció.

Példa:

```
procedure ConsolIO(command:byte;
                   var ch: byte;
                   error:  boolean);
begin
   reg.DL:=command;
   reg.AH:=$06;
   MSDos(reg);
   if (command=$ff) and (reg.Flags and 1)=1 then error:=true
   else begin
      ch:=reg.AL;
      error:=false
   end
end;
```

17.3.8 7-es funkció - közvetlen konzol input, képernyőre írás nélkül

A standard inputról vár egy karaktert, amelyet az AL-ben továbbít. A billentyű ellenőrzés ugyanaz, mint a 6-os funkciónál.

Kibővített ASCII kódok esetén két funkcióhívásra van szükség. Az első #0 kódú karakterrel tér vissza.

Példa:

```
function ReadConsole: byte;
begin
   reg.AH:=$07;
   MSDos(reg);
   ReadConsole:=reg.AL
end;
```

17.3.9 8-as funkció - konzol input képernyőre írás nélkül

Karaktert olvas be AL-be a standard inputról, mint a 7-es funkció, de nem írja vissza. <Ctrl-Break> megnyomása esetén a 23-as megszakítást is végrehajtja.

Példa:

```
function InpKbd: byte;
begin
   reg.AH:=$08;
   MSDos(reg);
   InpKbd:=reg.AL
end;
```

17.3.10 9-es funkció - karaktersorozat nyomtatása

Hívásakor a DS:DX egy olyan karaktersorozatra mutat a memóriában, amely dollárjellel ($: hexadecimális 24) fejeződik be. A sorozat minden jele a standard outputra kerül (a dollárjel már nem).

Ez a funkció egyik regiszterben sem ad vissza információt.

Példa:

```
procedure DisplayStr(irat: szoveg);
begin
   irat:=irat+'$';
   reg.DS:=Seg(irat);
   reg.DX:=Ofs(irat)+1;
   reg.AH:=$09;
   MSDos(reg)
end;
```

17.3.11 A funkció - pufferelt tasztatúra input

A DS:DX az input pufferra mutat. Az input puffer első byte-ja nem lehet nulla, hanem a pufferben maximálisan tárolható karakterek számát adja meg. A funkció a karaktereket a standard input egységről olvassa és a puffer 3. byte-jától kezdve helyezi el. A standard inputról az átvitel mindaddig folytatódik, amíg az <Enter> billentyűt le nem nyomjuk. Ha már csak egy karakter fér el a pufferbe és még további karaktereket akarunk olvasni, akkor karakterenként egy-egy hangjelzés figyelmeztet arra, hogy az <Enter>-t nyomjuk le, és ezeket a karaktereket nem olvassa be. A puffer második byte-ja a ténylegesen átvitt karakterek számát adja meg. Az "üzenet vége"-jel (carrige return 0D) nem tartozik a jelsorozathoz, ez mindig az utolsó jel. A puffer felépítését a későbbiekben ismertetjük.

Példa:

```
procedure ReadLine(MaxLength: byte;
                   var irat: szoveg);
begin
   irat[0]:=chr(MaxLength);
   reg.DS:=Seg(irat);
   reg.DX:=Ofs(irat);
   reg.AH:=$0a;
   MSDos(reg);
   irat:=copy(irat,2,ord(irat[1]))
end;
```

17.3.12 B funkció - standard input egység állapotának ellenőrzése

Ha a standard input egységről egy karakter átvihető, akkor (AL) = FF, különben (AL) = 00. <Ctrl-Break> esetén 23-as megszakítás történik.

Példa:

```
function IsEmptyKbd: boolean;
begin
   reg.AH:=$0b;
   MSDos(reg);
   if reg.AL=$ff then IsEmptyKbd:=false
   else IsEmptyKbd:=true
end;
```

17.3.13 C funkció - a standard input puffer törlése, és egy standard input kezelő funkció hívása

A standard input pufferbe bevitt valamennyi karakter törlődik, majd az AL-ben megadott sorszámú funkció kerül végrehajtásra. (Csak 1, 6, 7, 8 és A megengedett.) A rendszer addig vár, amíg legalább egy karaktert be nem olvas.

A funkció az alfunkciónak megfelelően a beolvasott karakter(eke)t vagy állapotot adja vissza.

Példa:

```
procedure FlushAndRead(command: byte;
                       var mode: byte);
begin
   reg.AL:=mode;
   reg.AH:=command;
   MSDos(reg);
   mode:=reg.AL
end;
```

17.3.14 D funkció - lemez alaphelyzetbe állítása

A funkció inicializálja a lemezegységet, a speciális file puffereket kiüríti. A módosított file-ok, melyeknek jellemzői (dátum, méret stb.) megváltoztak, hibásan maradnak a tartalomjegyzékben. Lemezcsere esetén, ha az összes olyan file-t, amelybe írtunk, lezártuk, nem kell hívnunk ezt a funkciót.

Ez a funkció egyik regiszterben sem ad vissza információt.

Példa:

```
procedure ResetDisk;
begin
   reg.AH:=$0d;
   MSDos(reg)
end;
```

17.3.15 E funkció - lemezegység kiválasztása

A DL-ben megadott meghajtót (0 = A, 1 = B stb.) teszi aktuális meghajtóvá. A meghajtók számát az AL-be írja (floppy egységek száma + fix lemezek száma). Ha a számítógép csak egy meghajtóval rendelkezik, a rendszer akkor is két meghajtót feltételez, az A: és a B: logikai meghajtókat. (A floppy egységek tényleges számának meghatározására használhatjuk az INT 11-et is, amellyel a BIOS állapítja meg az egységek jelenlétét.)

Példa:

```
procedure SetDisk(drive: byte;
                  var MaxDrive: byte);
begin
   reg.DL:=drive;
   reg.AH:=$0e;
   MSDos(reg);
```

```
   MaxDrive:=reg.AL
end;
```

17.3.16 F funkció - file megnyitása

A DS:DX egy olyan FCB-re mutat, amelyet meg kell nyitni. Az aktuális tartalomjegyzékben megkeresi a megadott file-t. Ha nem találta meg, akkor az AL-ben FF-et ad vissza. Ha megtalálta, akkor az AL-be 00-t ír és az FCB-t a következő módon tölti fel:

Ha a meghajtókód = 0 (aktuális meghajtó), akkor ezt az aktuális meghajtó kódjára állítja (1 = A, 2 = B stb.). Így az aktuális meghajtó anélkül változtatható, hogy az ezzel a file-lal végzendő műveleteket befolyásolná. Az FCB aktuális blokkot megadó mezőjét nullára állítja. A feldolgozandó rekord hosszát hexadecimális 80-ra állítja. Ha más hosszt akarunk, ezt az értéket nekünk kell megváltoztatnunk. Ha random file-kezelésre készülünk, az FCB-nek a random file-kezeléssel foglalkozó byte-jait is állítani kell (lásd az FCB részletes leírását). Az FCB esetleges változtatását a megnyitás után, de a többi művelet előtt kell végrehajtani.

Példa:

```
procedure OpenFile(var fcb: fcbSet;
                   var error: boolean);
begin
   reg.DS:=Seg(fcb)
   reg.DX:=Ofs(fcb);
   reg.AH:=$0f;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.17 10-es funkció - file lezárása

A file módosítása után végre kell hajtani, hogy helyes információk kerüljenek a tartalomjegyzékbe. A DS:DX egy megnyitott FCB-re mutat. A funkció a lemez aktuális tartalomjegyzékét átvizsgálja. Ha megtalálja a file-t, akkor helyzetét összehasonlítja a file megnyitásakor a tartalomjegyzékben talált helyzetinformációkkal. Ha ezek nem egyeznek, akkor feltételezi, hogy a floppy-t kicserélték. Ebben az esetben, vagy ha a file-t az aktuális tartalomjegyzékben nem találja, az AL-be FF-et ír, egyébként 00-t és módosítja is a lemez tartalomjegyzékét az FCB-ben talált információnak megfelelően.

Példa:

```
procedure CloseFile(fcb: fcbSet;
                    var error: boolean);begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
```

```
   reg.AH:=$10;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.18 11-es funkció - adott FCB-beli file-név keresése tartalomjegyzékben

A DS:DX-ben megadott, nem nyitott FCB-hoz a DOS megkeresi a file-név első előfordulását. A név tartalmazhat globális név-karaktereket ($ és *) is. Ezt a file-nevet aztán megpróbálja az aktuális meghajtó tartalomjegyzékében is megtalálni. Ha nem talál ilyent, AL-be FF-et tesz. Ha talál, AL-be 00-t tesz, és a megtalált file további adatait a lemez átviteli területre (DTA) olvassa.

Aszerint, hogy a DS:DX-ben megadott FCB kibővített, vagy normál, a DTA-ba sikeres keresés után a következő információ kerül:

Ha az FCB kibővített, akkor a 11-es funkció a DTA első byte-ját FF-re, az azt követő 5 byte-ot 0-ra, az ezt követőt a megadott FCB keresésében használt attributum byte-ra állítja, majd leírja (a szokott módon: 1 = A, 2 = B stb.) a meghajtót, ezután teszi a 32 byte-os tartalomjegyzék bejegyzést (directory entry). Így a DTA-ba egy érvényes kiterjesztett FCB kerül, amely egy nem megnyitott file-hoz tartozik, s ugyanaz az attributum byte-ja, mint a funkció számára megadott, keresett FCB-é.

Ha a keresésnél használt FCB normál, akkor az első byte a meghajtó száma (a szokott módon: 1 = A, 2 = B stb.), az ezt követő 32 byte a megtalált file-névhez tartozó tartalomjegyzék bejegyzés. Így a DTA-ba egy érvényes, normál, nem megnyitott FCB kerül.

Ha kibővített FCB-t adtunk meg, a keresést az attributum-byte értéke határozza meg:

1. Ha az FCB attributum byte-ja nulla, akkor a keresés a kötetazonosítót (volume label), az altartalomjegyzékeket, a rejtett és a rendszer-file-ok bejegyzéseit nem találja meg.

2. Ha az attributum-byte rejtett-, vagy rendszer-file-t jelez, akkor a funkció az egyező normál file-okon kívül ezeket is megtalálja. Ha minden egyező tartalomjegyzék bejegyzést meg akarunk kapni, kivéve a kötetazonosítót, az attributum-byte mindhárom bitjét 1-re kell állítani (lásd a lemez tartalomjegyzék leírását).

3. Ha az attributum-byte a kötetazonosítóra van állítva, akkor részleges keresés történik, és csak az egyező adathordozó-azonosítót találja meg.

Példa:

```
procedure FirstEntry(fcb: fcbSet;
                     var error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$11;
```

```
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.19 12-es funkció - a 11-es funkcióval megtalált file-t követő további egyező file keresése

Ha a 11-es funkció talált egy, a megadottal egyező bejegyzést, akkor a 12-es segítségével megkereshetjük a következő egyező bejegyzést. Megengedett a file-azonosítóban a globális név-jelek (? és *) használata. Az input és output regiszterek értéke azonos a 11-es funkcióéval. Az FCB fenntartott területén azok az információk állnak, amelyek a keresés folytatásához szükségesek. (Több file keresésénél ezt az FCB-t ne használjuk file-műveletekhez!)

Példa:

```
procedure NextEntry(fcb: fcbSet;
                    var error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$12;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.20 13-as funkció - file törlése

A DS:DX egy nem megnyitott FCB-re mutat. A DOS az aktuális tartalomjegyzékből valamennyi, a megadottal egyező bejegyzést törli. Ha nincs egyező, akkor az AL regiszter FF-et, különben 00-t tartalmaz. A megadott file nem lehet "READ ONLY" attribútumú. A file-név a ? globális név-jelet tartalmazhatja.

Példa:

```
procedure DeleteFile(fcb: fcbSet;
                     var error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$13;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.21 14-es funkció - file soros olvasása

A DS:DX egy megnyitott FCB-re mutat. Az ebben meghatározott aktuális blokkszámmal és aktuális rekordszámmal címzett rekordot (lásd az FCB leírását) a funkció a DTA-ba olvassa, és aztán az aktuális rekord számát eggyel növeli. Ez a soros, másnéven szekvenciális olvasás.

Az AL értéke 0, ha az olvasás sikerült; 01, ha az olvasás EOF-ot talált és adatot egyáltalán nem tudott beolvasni; 02, ha a DTA hossz nem volt elég az olvasáshoz (szegmenshatár átlépés lenne); 03, ha EOF-ot talált, és csonka rekordot olvasott.

Példa:

```
procedure SeqRead(fcb: fcbSet;
                  var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$14;
   MSDos(reg);
   mode:=reg.AL
end;
```

17.3.22 15-ös funkció - file soros írása

A soros, másnéven szekvenciális írás esetén is egy megnyitott FCB-re mutat a DS:DX. A rekord file-ba kerülési helyét az FCB ugyanúgy meghatározza, mint a 14-es funkciónál.

A rekord a DTA-ból kerül a file-ba. Ha a rekord rövidebb, mint egy szektor, akkor a funkció pufferolja a rekordokat, míg egy szektornyi összegyűlik, és blokkonként írja ki.

Ezután az FCB-ben a rekord sorszámát eggyel megnöveli.

Az AL értéke 0, ha az írás sikerült; 01, ha a lemez már tele volt - és ekkor írás egyáltalán nem történt; 02, ha a DTA túl kicsi, a teljes kiírandó blokk túlnyúlna a szegmenshatáron.

Példa:

```
procedure SeqWrite(fcb: fcbSet;
                   var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$15;
   MSDos(reg);
   mode:=reg.AL
end;
```

17.3.23 16-os funkció - file létrehozása

A DS:DX egy nem megnyitott FCB-re mutat. A lemez aktuális tartalomjegyzékében az adott tartalomjegyzék bejegyzéssel azonos bejegyzést keres a rendszer. Ha talál ilyet, akkor ezt a bejegyzést ismételten felhasználja. Ha nem talál, akkor a következő szabad bejegyzést megkeresi a tartalomjegyzékben. Ha ilyet nem talál, akkor az AL-ben FF-et ad vissza, különben a file-t megnyitja, és az AL-be 00-t tölt.

A funkcióval rejtett file-t is létre lehet hozni kibővített FCB-vel és a megfelelő attributum byte értékkel. Ha meglevő file bejegyzést talál a DOS, a file korábbi tartalma elvész.

Példa:

```
procedure NewFileName(fcb: fcbSet;
                      var error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$16;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.24 17-es funkció - file átnevezése

A DS:DX most egy olyan - módosított - nem megnyitott FCB-re mutat, amelyben a file-név (a régi) és a meghajtó kódja a szokott helyen áll, az új file-név pedig a régi file-név után 8 byte-tal (DS:DX + hexadecimális 11) kezdődik. A funkció, ha az aktuális tartalomjegyzékben az első file-névvel egyezőt talál, akkor annak a nevét a másodikra cseréli. (A két file azonosítója nem lehet ugyanaz.) A régi név azon pozícióiban álló karakterek, amelyeknek megfelelő helyen a másodikban "?" szerepel, változatlanul kerülnek át az új névbe.

Ha a funkció a megadott régi névvel egyező nevű file-t nem talál vagy az új név már foglalt, AL-t FF-re állítja.

Példa:

```
procedure NewFile(fcb: fcbSet;
                  var error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$17;
   MSDos(reg);
   if reg.AL=$ff then error:=true
```

```
   else error:=false
end;
```

17.3.25 18-as funkció - a DOS számára fenntartott funkció

17.3.26 19-es funkció - az aktuális meghajtó kódjának bekérése

A funkció AL-ben az aktuális meghajtó kódját adja vissza. (0 = A, 1 = B stb.) Hívási paramétere nincs.

Példa:

```
function GetDisk: byte;
begin
   reg.AH:=$19;
   MSDos(reg);
   GetDisk:=reg.AL
end;
```

17.3.27 1A funkció - a DTA címének beállítása

A lemez átviteli terület (DTA) címének beállítása. A kívánt címet DS:DX-ben adjuk meg. A lemez átviteli terület maximális mérete egy szegmens (64 Kbyte). A DTA nem nyúlhat át a szegmenshatáron.

Példa:

```
procedure SetDTA(cim: integer2);
begin
   reg.DS:=cim.HighInt;
   reg.DX:=cim.LowInt;
   reg.AH:=$1a;
   MSDos(reg)
end;
```

17.3.28 1B funkció - az aktuális meghajtó információinak bekérése

A funkció segítségével a FAT (file-allokációs táblázat) információit kaphatjuk meg. A funkció hívás után a DS:DX az aktuális meghajtó FAT-jának első byte-jára mutat, amely a meghajtó típusát azonosítja. BX a lemezen lévő cluster-ek számát (lásd a FAT-tal foglalkozó szakaszt), AL a szektorok számát cluster-enként, CX a fizikai szektorméretet tartalmazza. Hívási paramétere nincs.

Példa:

```
procedure GetDeafaultDiskData(var SektorPerCluster: byte;
                              var BytesPerSector: integer;
                              var ClusterPerDrive: integer;
                              var TypeByte: byte);
begin
   reg.AH:=$1b;
   MSDos(reg);
   SectorPerCluster:=reg.AL;
   BytesPerSector:=reg.CX;
   ClusterPerDrive:=reg.DX;
   TypeByte:=mem[reg.DS,reg.BX]
end;
```

17.3.29 1C funkció - adott meghajtó információinak bekérése

A funkció hívásakor a kívánt meghajtó számát DL-ben kell megadni (a szokásos módon: 0 az aktuális, 1 az A meghajtónak stb. felel meg).

A funkció outputja ugyanaz, mint az 1B funkciónál.

Példa:

```
procedure GetDeafaultDiskData(drive: byte;
                              var SektorPerCluster: byte;
                              var BytesPerSector: integer;
                              var ClusterPerDrive: integer;
                              var TypeByte: byte);
begin
   reg.DL:=drive;
   reg.AH:=$1c;
   MSDos(reg);
   SectorPerCluster:=reg.AL;
   BytesPerSector:=reg.CX;
   ClusterPerDrive:=reg.DX;
   TypeByte:=mem[reg.DS,reg.BX]
end;
```

17.3.30 1D-20 funkciók - a DOS számára fenntartott funkciók

17.3.31 21-es funkció - random olvasás

A random (közvetlen) olvasási funkció inputként DS:DX-ben egy megnyitott FCB címét várja. Az FCB aktuális blokk és aktuális rekord mezőit megfelelően be kell állítani. Az ezekkel a mezőkkel címzett rekord a memóriában az aktuális DTA-ba kerül.

Ha az olvasás sikeresen befejeződött, a funkció AL-t 0-ra állítja. Ha file-véget (EOF) talál, akkor az AL-be 01-et vagy 03-at tesz: 01 esetén nem állt rendelkezésre több adat, 03 esetén csonka rekordot olvasott be, és a hiányzó részt 0-val töltötte fel. AL = 02 lesz, ha a DTA túl kicsi a rekord fogadására, ekkor olvasás egyáltalán nem történik.

Példa:

```
procedure RandomRead(fcb: fcbSet;
                     var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$21;
   MSDos(reg);
   mode:=reg.AL
end;
```

17.3.32 22-es funkció - random írás

A random (közvetlen) írási funkció inputként DS:DX-ben egy megnyitott FCB címét várja. Az FCB aktuális blokk és aktuális rekord mezőit a megfelelően ki kell tölteni. Az ezekkel a mezőkkel címzett területről írja a file-ba a rekordot a DTA-ból. (Ha az adatrekord mérete nem egyezik meg a szektor méretével, akkor pufferolt I/O műveletet hajt végre a rendszer.)

Az AL-ben visszaadott értékek jelentése:

00 - Sikeres átvitel.
01 - A lemez tele van, az írás nem történt meg.
02 - A DTA szegmenshatáron átnyúlik. Az átvitel nem történt meg.

Példa:

```
procedure RandomWrite(fcb: fcbSet;
                      var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$21;
   MSDos(reg);
   mode:=reg.AL
end;
```

17.3.33 23-as funkció - file-méret lekérdezése

A funkcióba lépéskor DS:DX egy nem megnyitott FCB-re mutat.

A funkció a lemez aktuális tartalomjegyzékében megkeresi a file-azonosítóval egyező első

bejegyzést. Ha nem talál ilyet, akkor AL-ben FF-et ad vissza. Különben az FCB "random rekord" mezőibe beírja a file rekordjainak számát (az FCB "rekord méret" mezejében kifejezve, kerekítve), és AL-t 0-ra állítja.

Megjegyzés:

Mielőtt ezt a hívást kiadjuk, az FCB "rekord méret" mezőjét be kell állítani, különben hibás információt kapunk. (Ha a file méretére vagyunk kíváncsiak, a mezőbe 1 értéket kell írni.)

Példa:

```
procedure GetFileSize(fcb: fcbSet;
                      error: boolean);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$23;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.34 24-es funkció - random rekord mező beállítása

A DS:DX belépéskor egy megnyitott FCB-re mutat. Ez a hívás a random rekord mezőt (fcb.RelRecord) az FCB aktuális blokk és rekord mezőinek (fcb.CurrBlock, fcb.RecordSize) megfelelően állítja be.

Megjegyzés:

Random írási/olvasási (lásd 21, 22 funkciók) és random blokkírási/olvasási (lásd 27, 28 funkciók) műveletek előtt ezzel a funkcióval kell a random rekord mezőt előkészíteni.

Példa:

```
procedure SetRecord(fcb: fcbSet);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.AH:=$24;
   MSDos(reg)
end;
```

17.3.35 25-ös funkció - a megszakítási (interrupt) vektor beállítása

Inputként a funkció AL-ben a megszakítás számát, DS:DX-ben az interrupt kezelő rutin címét várja.

Regiszter-outputja a funkciónak nincs. Az adott számú interrupt-hoz az interrupt vektor táblában az adott rutincímet rendeli hozzá. (Az eredeti táblabeli érték a 35-ös funkcióval kérdezhető le.)

Példa:

```
procedure SetVector(number: byte;
                    cim: integer2);
begin
   reg.AL:=number;
   reg.DS:=cim.HighInt;
   reg.DX:=cim.LowInt;
   reg.AH:=$25;
   MSDos(reg)
end;
```

17.3.36 26-os funkció - új program szegmens előállítása

A DX belépéskor az új program szegmens szegmenscímét (paragrafusban számolva) tartalmazza.

A funkció az aktuális program szegmens 0-s címétől hexadecimális 100 hosszú területet átmásol az új szegmens 0-s címétől kezdődő területre. A 0A címtől kezdve beírja az aktuális befejezési, a Ctrl-Break kilépési-, és a kritikus hiba címeket (az interrupt vektor tábla 22-es, 23-as és 24-es megszakításokhoz tartozó értékei alapján), és az új szegmens 6-os címére beírja az új memóriaméretet.

Megjegyzés:

E funkció helyett a 4B (EXEC) funkcióhívás az ajánlott.

17.3.37 27-es funkció - random blokk olvasása

Belépéskor a DS:DX egy megnyitott FCB-re mutat. A CX az olvasandó rekordok számát tartalmazza és nem lehet nulla. A funkció a megadott számú rekordot a random rekord mezőben megadott file címről a lemez átviteli területre (DTA) olvassa.

A funkció outputja:

Ha file-vég (EOF) következik be, akkor az AL-ben 01-et vagy 03-at ad vissza aszerint, hogy nem olvasott adatot (01) vagy csonka rekordot olvasott (03). Utóbbi esetben a hiányzó részt 0-val tölti ki. Ha a lemez átviteli területre (DTA) egy olyan rekordot tölt, amelyik már a szegmenshatárt túllépné, akkor az AL-be 02 kerül, de a rendelkezésre álló területet még teljesen feltölti.

Ha valamennyi rekordot sikeresen beolvasta, akkor AL-ben 00-t ad vissza. CX mindig a ténylegesen beolvasott rekordok számát tartalmazza. Az FCB random, aktuális blokk és aktuális rekord mezőjét módosítja úgy, hogy a következő olvasandó rekordra mutassanak.

Példa:

```
procedure RandomBlockRead(fcb: fcbSet;
                          var counter: integer;
                          var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.CX:=counter;
   reg.AH:=$27;
   MSDos(reg);
   mode:=reg.AL;
   counter:=reg.CX
end;
```

17.3.38 28-as funkció - random blokk írása

Ugyanúgy működik, mint a 27-es funkció, csak itt írásról van szó, azaz DS:DX az FCB-t, CX az (írandó) rekordszámot adja meg.

Ha nincs elég hely a lemezen, akkor az AL-ben 01-et ad vissza és írás nem történik. Ha AL-t 02-re állítja, a DTA volt túl kicsi és írás szintén nem történt. AL=00-t kapunk sikeres írás esetén.

Ha belépéskor a CX nullát tartalmaz, akkor írás nem történik, a file méretét viszont aktualizálja a tartalomjegyzékben a random rekord mezőbeli méret szerint.

Példa:

```
procedure RandomBlockWrite(fcb: fcbSet;
                           var counter: integer;
                           var mode: byte);
begin
   reg.DS:=Seg(fcb);
   reg.DX:=Ofs(fcb);
   reg.CX:=counter;
   reg.AH:=$28;
   MSDos(reg);
   mode:=reg.AL;
   counter:=reg.CX
end;
```

17.3.39 29-es funkció - file-azonosító keresése adott parancssorban

Belépéskor DS:SI az elemzendő parancs-sorra mutat, ES:DI pedig egy meg nem nyitott FCB-re. Az AL tartalmától függően kerül a művelet végrehajtásra a következőképpen:

Ha a 0. bit = 1, akkor a parancssor vezető elválasztó jelei a parancssorban figyelmen kívül maradnak (különben nem).

Ha az 1. bit = 0, akkor a parancssorból a meghajtó azonosítója átkerül az FCB-be, amennyiben van ilyen a parancssorban.

Ha 2. bit = 1, akkor a file-név a parancssorból átkerül az FCB-be, ha az tartalmaz file-nevet.

Ha 3. bit = 1, akkor ez ugyanaz, mint az előbbi, csak a file-név kiterjesztésére vonatkozik.

A 4-7 bitek tartalma közömbös.

A funkció outputja:

DS:SI a vizsgált file-azonosító utáni első karakterre mutat, ES:DI a megváltoztatott FCB-re mutat, AL tartalma pedig:

00, ha a parancssorban nem szerepeltek globális file-név karakterek,
01, ha szerepeltek,
FF, ha a meghajtó megadás rossz volt.

Példa:

```
procedure ParseFile(fcb: fcbSet;
                    var command: byte;
                    var hely: mutato);
begin
   reg.ES:=Seg(fcb);
   reg.DI:=Ofs(fcb);
   reg.DS:=Seg(hely|);
   reg.SI:=Ofs(hely|);
   reg.AL:=command;
   reg.AH:=$29;
   MSDos(reg);
   command:=reg.AL;
   hely:=ptr(rg.DS,reg.SI)
end;
```

17.3.40 2A funkció - dátum lekérdezése

AL binárisan a napot (0 vasárnap, 1 hétfő stb.), CX az évet (1980 és 2099 között), DH a hónapot (1 és 12 között), DL a napot (1 és 31 között) adja meg.

Példa:

```
procedure GetDate(var year,month,day,DayOfWeek: integer);
begin
   reg.AH:=$2a;
   MSDos(reg);
   year:=reg.CX;
   month:=reg.DH;
   day:=reg.DL;
   DayOfWeek:=reg.AL
end;
```

17.3.41 2B funkció - dátum beállítása

A funkció inputként CX-ben, DH-ban és DL-ben úgy várja a beállítandó dátumot, ahogy azt a 2A funkció leírásában megadtuk.

Outputként AL-ben 0-t ad, ha átírta a dátumot, FF-et, ha érvénytelen dátumot adtunk meg.

Példa:

```
procedure SetDate(year,month,day: integer;
                  var error:boolean);
begin
   reg.AH:=$2b;
   reg.CX:=year;
   reg.DH:=month;
   reg.DL:=day;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.42 2C funkció - rendszeridő lekérdezése

A funkció outputja:

CH az órát (0-23), CL a percet (0-59), DH a másodpercet (0-59) és DL a századmásodpercet (0-99) tartalmazza.

Példa:

```
procedure GetTime(var hour,minute,second,hundredth: integer);
begin
   reg.AH:=$2c;
   MSDos(reg);
   hour:=reg.CH;
   minute:=reg.CL;
   second:=reg.DH;
   hundredth:=reg.DL
end;
```

17.3.43 2D funkció - rendszeridő beállítása

A funkció inputját azokból a regiszterekből, és olyan formában várja, ahogy azt a 2C funkciónál leírtuk.

Ha az időmegadás érvénytelen, akkor az AL-ben FF-et, ha érvényes, akkor 00-t ad vissza (és átállítja a rendszeridőt).

Példa:

```
procedure SetTime(hour,minute,second,hundredth: integer
                  error: boolean);
begin
   reg.CH:=hour;
   reg.CL:=minute;
   reg.DH:=second;
   reg.DL:=hundredth;
   reg.AH:=$2d;
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else error:=false
end;
```

17.3.44 2E funkció - VERIFY (lemezírás utáni ellenőrzés) kapcsoló beállítása

A funkció inputja:

AL-nek az ellenőrzés bekapcsolásakor 01-et, kikapcsolásakor 00-t kell tartalmaznia. Ha az ellenőrzés be van kapcsolva, akkor a lemezre történő íráskor ellenőrző visszaolvasást is végez a rendszer. Kritikus file-ok írásakor célszerű használni. A kapcsoló aktuális állása az 54-es funkcióval kérdezhető le.

Példa:

```
procedure SetVerify(state: byte);
begin
   reg.AL:=state;
   reg.AH:=$2e;
   MSDos(reg)
end;
```

17.3.45 2F funkció - lemez átviteli terület (DTA) címének lekérdezése

A funkció outputja:

ES:BX az aktuális lemez átviteli terület (DTA) címét tatalmazza.

Példa:

```
procedure GetDTA(cim: integer2);
begin
   reg.AH:=$2e;
   MSDos(reg);
   cim.HighInt:=reg.ES;
   cim.LowInt:=reg.BX
end;
```

17.3.46 30-as funkció - DOS verziószám lekérdezése

A funkció outputja:

AL a főverziószámot, AH a főverzión belüli változat értékét tartalmazza. (Például a 2.10 verzió esetén az AX-ben 0A02-t ad vissza a funkcióhívás.) A funkció PC-DOS verziónál nullázza a BX és CX regisztereket.

Megjegyzés:

Ha az AL 00-t tartalmaz, akkor ez azt jelenti, hogy egy, a DOS 2.00-t megelőző verziót használunk.

Példa:

```
function GetVersion: real;
begin
   reg.AH:=$2e;
   MSDos(reg);
   GetVersion:=reg.AL+reg.AH/100
end;
```

17.3.47 31-es funkció - programfutás befejezése úgy, hogy a program a memóriában marad

A funkció inputja:

AL egy bináris befejezési kódot, a DX a program számára lefoglalandó memóriaméretet tartalmazza paragrafusokban kifejezve. Ez a funkció befejezi az aktuális programot és a program kezdeti helyfoglalási igényét megpróbálja a DX-ben megadott számra átállítani.

Az AL-ben visszaadott befejezési kódot a programot hívó a WAIT funkcióhívással (4D), vagy (BATCH feldolgozásnál) az ERRORLEVEL alparanccsal ellenőrizheti.

Megjegyzés:

A befejezési kód felépítését lásd a 4D funkció ismertetésénél.

Példa:

```
procedure KeepProcess(ReturnCode: byte;
                      MemorySize: integer);
begin
   reg.AL:=ReturnCode;
   reg.DX:=MemorySize;
   reg.AH:=$31;
   MSDos(reg)
end;
```

17.3.48 32-es funkció - DOS számára fenntartott funkció

17.3.49 33-as funkció - Ctrl-Break lekérdezése és beállítása

A funkció inputja:

AL = 00 az aktuális állapot lekérdezését, AL = 01 az állapot beállítását jelenti. Ekkor DL = 00 a Ctrl-Break ellenőrzést kikapcsolja (OFF), DL = 01 pedig bekapcsolja (ON).

A funkció outputja:

Állapotlekérdezés esetén DL-ben jelzi vissza a ki-, illetve bekapcsolt állapotot, az előző értékek szerint.

Példa:

```
procedure CheckCtrlC(var SetBit, state: byte;
                     var error: boolean);
begin
   reg.AL:=SetBit;
   reg.DL:=state;
   reg.AH:=$33;
```

```
   MSDos(reg);
   if reg.AL=$ff then error:=true
   else
      begin
         error:=false;
         state:=reg.DL
      end
end;
```

17.3.50 34-es funkció - DOS számára fenntartott funkció

17.3.51 35-ös funkció - megszakítási (interrupt) vektor lekérdezése

Belépéskor az AL a megszakítás hexadecimális számát tartalmazza. A megadott megszakítást kezelő rutin címét a funkció ES:BX-ben adja vissza.

A rutin címét a 25-ös funkcióval lehet beállítani.

Példa:

```
procedure GetVector(number: byte;
                    cim: integer2);
begin
   reg.AL:=number;
   reg.AH:=$35;
   MSDos(reg);
   cim.HighInt:=reg.ES;
   cim.LowInt:=reg.BX
end;
```

17.3.52 36-os funkció - szabad lemezterület lekérdezése

Belépéskor DL a meghajtó sorszámát tartalmazza: 0=aktuális, 1=A stb. Ha a meghajtószám érvénytelen volt, akkor visszatéréskor az AX-be FFFF kerül, különben:

- BX a rendelkezésre álló lemezblokkok (cluster) számát,
- DX a meghajtón lévő összes lemezblokk számát,
- CX a szektoronkénti byte-ok számát,
- AX pedig clusterenkénti szektorok számát tartalmazza.

Megjegyzés:

DX, AL, CX ugyanezeket adja meg az 1B funkció hívása esetén is.

Példa:

```
procedure GetDiskSpace(drive: byte;
```

```
                        var FreeClusters,
                            BytesPerSector,
                            SectorsPerCluster,
                            ClustersPerDrive: integer;
                        var error: boolean);
begin
   reg.DL:=drive;
   reg.AH:=$36;
   MSDos(reg);
   if reg.AX:=$ffff then error:=true
   else
      begin
         FreeClusters:=reg.BX;
         BytesPerSector:=reg.CX;
         SectorsPerCluster:=reg.CX;
         ClustersPerDrive:=reg.AX;
         error:=false
      end
end;
```

17.3.53 37-es funkció - DOS számára fenntartott funkció

17.3.54 38-as funkció - ország-specifikus információk lekérdezése

Belépéskor DS:DX arra a 32 byte-os memóriaterületre mutat, ahová az ország specifikus információk kerülnek. AL vagy BX tartalmazza az ország kódját. Ha ez kisebb, mint 255, akkor AL, különben BX tartalmazza az ország kódját. Az utóbbi esetben AL értékét 255-re kell beállítani.(A DOS 2.00-ban AL-nek 00-nak kell lennie!)

A 32 byte-os puffer terület az alábbi infomációkat tartalmazza:

1 szó	A dátum és idő formáját jelző kód,
5 byte	ASCII-jelsorozat, a pénzegység jele, jobbról 0-val zárva,
2 byte	ASCII-jelsorozat, az ezresek elválasztásának jele, jobbról 0-val lezárva,
2 byte	ASCII-jelsorozat, a tizedes elválasztójel (pont, illetve vessző), jobbról 0-val lezárva,
2 byte	ASCII-jelsorozat, a dátum elválasztó jele, jobbról 0-val lezárva,
2 byte	ASCII-jelsorozat, az idő elválasztó jele, jobbról 0-val lezárva,
1 byte	Jelző bitek,
1 byte	Jelző: pénzegység használatakor a tizedes jegyek száma,
1 byte	Jelző: idő formátuma.
1 duplaszó	A kisbetűk/nagybetűk átalakító rutin kezdőcíme,
2 byte	ASCII-jelsorozat, az adatelválasztó jel jobbról 0-val lezárva,
10 byte	Fenntartott (a DOS használja).

A dátum és idő megadása formáját jelző szó értékei a következők lehetnek:

A szó értéke:	A szabványhoz tartozó dátum-megadás
0 = USA	szabvány: hónap nap év
1 = európai	szabvány: nap hónap év
2 = japán	szabvány: év hónap nap

A Jelzőbitek jelentése:

0. bit	= 0 a pénzegység jele a szám előtt,
	= 1 a pénzegység jele a szám után,
1. bit	= 0 nincs szóköz a pénzegység és a szám között,
	= 1 szóköz a pénzegység és a szám között.

Az idő formátum byte 0. bitjének leheséges értékei:

0 = 12 órás kijelzés,
1 = 24 órás kijelzés.

Példa:

```
procedure GetCountryData(CountryCode: integer;
                         var puffer: mutato;
                         var error: boolean);
begin
   if CountryCode<255 then reg.AL:=CountryCode
   else begin
      reg.AL:=255; reg.BX:=CountryCode
   end;
   reg.DS:=Seg(puffer^);
   reg.DX:=Ofs(puffer^);
   reg.AH:=$38;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=true
   else error:=false
end;
```

17.3.55 39-es funkció - altartalomjegyzék létrehozása

Belépéskor DS:DX a meghajtót és elérési utat leíró ASCII jelsorozat címét tatalmazza.

Ha az elérési út valamelyik eleme nem létezik, akkor az altartalomjegyzék nem jön létre (Carry Yes), egyébként a megadott elérési út végére írja fel az új altartalomjegyzéket (No Carry).

Példa:

```
procedure MakeDir(pathname: MaxString;
                  var error: boolean);
```

```
begin
   pathname:=pathname+chr(0);
   reg.DS:=Seg(pathname);
   reg.DX:=Ofs(pathname);
   reg.AH:=$39;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=true
   else error:=false
end;
```

17.3.56 3A funkció - altartalomjegyzék törlése

Belépéskor DS:DX egy olyan memóriaterületre mutat, ahol a meghajtót és elérési utat leíró ASCII-jelsorozat van.

Az így megadott altartalomjegyzék a struktúrából törlődik. Az aktuális tartalomjegyzék nem törölhető.

Ha a funkció végrehajtása után Carry = Yes, akkor az AX-beli érték a hibakódot jelenti.

Példa:

```
procedure DelDir(pathname: MaxString;
                 var error: boolean);
begin
   pathname:=pathname+chr(0);
   reg.DS:=Seg(pathname);
   reg.DX:=Ofs(pathname);
   reg.AH:=$3a;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=true
   else error:=false
end;
```

17.3.57 3B funkció - aktuális alkönyvtár beállítása

Belépéskor DS:DX egy olyan memóriaterületre mutat, ahol a meghajtót és elérési utat leíró jelsorozat van.

Ha az elérési út valamely eleme nem létezik, nem történik változtatás (Carry Yes), különben az aktuális altartalomjegyzék a megadott lesz (No Carry).

Példa:

```
procedure SetDir(pathname: MaxString;
                 var error: boolean);
begin
   pathname:=pathname+chr(0);
```

```
   reg.DS:=Seg(pathname);
   reg.DX:=Ofs(pathname);
   reg.AH:=$39;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=true
   else error:=false
end;
```

17.3.58 3C funkció - file létrehozása vagy meglévő file hosszának 0-ra állítása

Belépéskor DS:DX egy olyan memóriaterületre mutat, ahol a meghajtót, elérési utat és a file-azonosítót leíró ASCII-jelsorozat van, CX tartalmazza a file attributumot.

Ez a funkcióhívás egy új file-t állít elő vagy egy meglévő file-t 0-hosszúságúra állít. Ha a file nem létezik, akkor a megadott tartalomjegyzékbe bejegyzi. A file-hoz írási-olvasási elérési kódot rendel, és a file-t megnyitja. Ha a funkció rendben végrehajtódott, akkor a carry nincs beállítva, és AX-ben a file-számot nyerjük vissza.

Megjegyzés:

A file attributuma a 43-as funkcióhívással változtatható meg.

Példa:

```
procedure CreateFile(filename: MaxString;
                     attribute: byte;
                     var error: boolean);
begin
   fatname:=filename+chr(0);
   reg.DS:=Seg(filename);
   reg.DX:=Ofs(filename);
   reg.CX:=attribute;
   reg.AH:=$39;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=true
   else error:=false
end;
```

17.3.59 3D funkció - file megnyitása

Belépéskor DS:DX egy komplett file-specifikációra mutat, AL az elérési kódot tartalmazza.

Az elérési kód megengedett értékei a DOS 2.11-3.00 változataiban:

0 = file megnyitása olvasásra,
1 = file megnyitása írásra,
2 = file megnyitása írásra/olvasásra.

A DOS 3.10-es verzióitól kezdve a kód byte bitjeinek a következő a jelentése:

7. bit: öröklési bit,
6.-4. bitek: megnyitási mód,
3.-0. bitek: elérési mód.

Az öröklési bit jelzi, hogy a file átadandó-e egy, a processzor által elindított másik processzornak is. Ha a bit magas, az új processzor nem 'örökli' a file-t.

A megnyitási mód 4.-6. bitjeit a rendszer a következőképpen használja:

000 Kompatibilis mód. A file-t tetszőleges processzor megnyithatja, tetszőleges sokszor. Hibát jelez, ha bármilyen más módon próbálják megnyitni.

001 Teljes lefoglalás. A file-t ezután már semmilyen módon sem lehet megnyitni. Újra megnyitására még a megnyitó processzora sem képes. (Előtte le kell zárni a file-t)

010 Lefoglalás írásra. A file nem lesz megnyitva, ha egy másik processzor megnyitotta kompatibilis módban, vagy lefoglalta írásra.

011 Lefoglalás olvasásra. A file nem lesz megnyitva, ha egy másik processzor megnyitotta kompatibilis módban, vagy lefoglalta olvasásra.

100 Lefoglalás. A file nem lesz megnyitva, ha bármely más processzor megnyitotta kompatibilis módban.

Az elérési módok a következők:

0000 Olvasás. A file nem nyitható meg, ha már teljesen vagy olvasásra legfoglalták.
0001 Írás. A file nem nyitható meg, ha már teljesen vagy írásra legfoglalták.
0010 Mindkettő. A file nem nyiható meg, ha már teljesen, írásra vagy olvasásra lefoglalták.

Visszatéréskor az AX a hibakódot vagy a 16 bites file-számot tartalmazza aszerint, hogy a carry-t beállítva (hibakód) vagy nem beállítva (file-szám) kapjuk vissza.

A lehetséges hibák:

2 = a megadott file nem használható vagy nem létezik;
3 = a megadott út érvénytelen vagy nem létezik;
4 = a processzor számára nincs elérhető file-szám, vagy megtelt a DOS belső táblázata;
5 = a program egy altartalomjegyzéket, egy kötetazonosítót vagy egy csak-olvasható file-t próbált megnyitni;
12 = az elérési kód nem esik a 0-2 intervallumba.

A funkcióból való kilépés után az írás/olvasás a file első byte-jától kezdődhet, ide mutat az írás/olvasás pointer. Az aktuális rekordhossz 1 byte lesz. A pointer értékét a 42-es funkcióhívással módosíthatjuk, a rekordméretet olvasásnál a 3F, írásnál a 40-es funkciónál kell megadni.

A file utolsó megnyitásának dátumát és idejét az 57-es funkcióval állíthatjuk be vagy kérdezhetjük le, attributumát a 43-as funkcióval lehet lekérdezni, ill. módosítani.

Példa:

```
procedure Openfile(filename: MaxString;
                       mode: byte;
                       var filehandle: integer;
                       var error:integer);
begin
   filename:=filename+chr(0);
   reg.DS:=Seg(filename);
   reg.DX:=Ofs(filename);
   reg.AL:=mode;
   reg.AH:=$3D;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=reg.AX
   else begin
      error:=0;
      filehandle:=reg.AX
   end
end;
```

17.3.60 3E funkció - file lezárása

A funkció a BX-ben megadott file-számú file-t zárja le (ezt a számot a 3C vagy 3D funkciók adták meg AX-ben).

A funkció a lezáráson kívül a tartalomjegyzék aktualizálását, és a file-hoz tartozó pufferek kiürítését is elvégzi. Hibajelzés: Carry bitben.

Példa:

```
procedure CloseFile(filehandle: integer;
                       var error:integer);
begin
   reg.BX:=filehandle;
   reg.AH:=$3E;
   MSDos(reg);
   if (reg.Flags and 1)=1 then error:=reg.AX
   else error:=0
end;
```

17.3.61 3F funkció - olvasás file-ból vagy egyéb perifériáról

A funkció a BX-ben megadott file-számú file-ból CX-ben megadott számú byte-ot olvas a DS:DX-ben megadott kezdőcímű pufferba.

Visszatéréskor a funkció AX-ben adja meg a ténylegesen olvasott byte-ok számát vagy sikertelen olvasás esetén (Carry Yes) a hibakódot.

Nem feltétlenül a CX-ben megadott számú byte-ot olvas a funkció, ugyanis például a klaviatúráról legfeljebb egy sornyi szöveg jöhet egyszerre, ill. ha a file végéig már kevesebb byte van, mint az adott szám, akkor annyit olvas, amennyi még rendelkezésre áll.

A standard inputról való olvasás átirányítható (lásd "A PC DOS Speciális lehetőségei" című 12. fejezetben a megfelelő szakaszt).

Ha AX-ben 0-t kapunk, a file végén túl kíséreltünk meg olvasni.

Példa:

```
procedure ReadHandle(handle:integer;
                     var count: integer;
                     var mode: integer);
begin
   reg.BX:=handle;
   reg.CX:=count;
   reg.DS:=Seg(puffer);
   reg.DX:=Ofs(puffer);
   reg.AH:=$3f;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else begin
      mode:=0;
      count:=reg.AX;
   end;
end;
```

17.3.62 40-es funkció - írás file-ba vagy egyéb perifériára

A funkció a BX-ben megadott file-számú file-ba a CX-ben megadott számú byte-ot ír a DS:DX-ben megadott című pufferből.

Akár e funkció olvasó párja, a 3F-es, ez is AX-ben adja vissza a ténylegesen írt byte-ok számát. Itt azonban ha (CX) = (AX), akkor hiba történt (például már nincs a lemezen hely), így programunkban ezt az esetet célszerű ellenőrizni.

Ha az írás még ennyire sem volt sikeres (Carry Yes), a hibakódot kapjuk AX-ben.

A standard output ugyanúgy átirányítható, mint a 3F funkció esetén az input.

17.3.63 41-es funkció - file törlése

Belépéskor DS:DX egy komplett file-specifikáció címét tartalmazza. A globális file-név karakterek használata nem megengedett.

A funkció a megadott file-t törli. Csak olvasható file-t a funkció nem tud törölni, ilyenkor a file attributumát - a funkció hívása előtt - a 43-as funkcióval 0-ra kell állítani.

Példa:

```
procedure DeleteDirectory entry(path:string64;
                                var mode:integer);
begin
   path:=path+chr(0);
   reg.DS:=Seg(path);
   reg.DX:=Ofs(path);
   reg.AH:=$41;
   MSDos(reg);
   if (reg.Flags and 1)=0 then mode:=0
   else mode:=reg.AX
end;
```

17.3.64 42-es funkció - írási/olvasási mutató átállítása

Amikor egy file-t a 3D funkcióval megnyitunk, az írási/olvasási mutató a file első byte-jára mutat.

A file feldolgozása során ezt a mutatót mindig be kell állítani, ha nem sorosan olvasunk/írunk.

Belépéskor BX tartalmazza a kezelendő file számát, CX:DX azt az értéket, hogy a mutató kurrens értéke az AL-ben meghatározott módon számítva hány byte-tal módosuljon - ez az érték az ún. eltolási vagy offset érték. (CX tartalmazza a magasabb helyiértékű részt.)

Az eltolást AL a következőképpen határozza meg:

0 = a file elejétől,
1 = a mutató jelenlegi állásától,
2 = a file végétől visszafelé.

A funkció output-ként DX:AX-ben adja meg a mutató új helyzetét, ha a carry nem volt beállítva. (DX-ben van a magasabb helyiértékű rész.)

Példa:

```
procedure MovePointer(handle:integer;
                      var distance: integer2;
                      method: byte;
                      var mode: integer);
begin
   reg.AL:=method;
   reg.BX:=handle;
   reg.CX:=distance.High;
```

```
   reg.DX:=distance.Low;
   reg.AH:=$42;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=1
   else begin
      mode:=0;
      distance.High:=reg.DX;
      distance.Low:=reg.AX;
   end;
end;
```

17.3.65 43-as funkció - file attributumának lekérdezése vagy módosítása

Ha belépéskor (AL) = 0, a funkció lekérdezi, ha (AL) = 01, módosítja a file attributumát a következők szerint:

DS:DX egy komplett file-specifikáció címét tartalmazza, CX-ben adjuk, illetve kapjuk meg a file attributumát.

Megjegyzés:

A hexadecimális 08 és 10 attributum (kötetazonosító, illetve altartalomjegyzék jelzés) nem változtatható meg.

Példa:

```
procedure FileAttribute(file name:string64;
                        command: byte;
                        var attributes: byte;
                        var mode: integer);
begin
   file name+chr(0);
   reg.DS:=Seg(file name);
   reg.DX:=Ofs(file name);
   if command=1 then reg.CX:=attributes;
   reg.AL:=command;
   reg.AH:=$43;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else begin
      mode:=0;
      if command=0 then attributes:=reg.CX;
   end;
end;
```

17.3.66 44-es funkció - periféria vagy file-leíró információ lekérdezése vagy módosítása

A funkció inputja:

Az AL regiszter tartalmával határozzuk meg, hogy a sok lehetőség közül a funkció melyik feladatot lássa el.

DS:DX egy adat vagy puffer címe.
CX az írandó vagy olvasandó byte-ok száma (AL = 02,03,04 vagy 05 alfunkció esetén).
BX a file-szám (AL = 00,06,07,0A,0C esetén).
BL a meghajtó azonosítója (AL = 04,05,08,09,0C,0E vagy 0F esetén).

A DOS eltérő verzióiban nem mindegyik alfunkció használható:

DOS -2.11:	00H-07H
DOS 3.0 és 3.1:	00H-0BH
DOS 3.2-:	00H-0FH

A funkció a következő feladatok ellátására képes:

AL = 00 periféria leíró információ lekérdezése DX-be - az információ részletezése majd következik,

AL = 01 periféria leíró információ átállítása, amelyet DX-ben adunk meg. DL-ben a táblázat 7-0. bitje módosítható.),

AL = 02 (CX) számú byte olvasása DS:DX-be a periféria vezérlő csatornájáról. (BX) adja meg a perifériát,

AL = 03 ugyanennek az írása, lásd AL = 02,

AL = 04 ugyanaz, mint 02, csak nem a file-számmal kijelölt perifériára, hanem a (BL)-ben a meghajtó-azonosítóval adjuk meg a perifériát,

AL = 05 a 03 megfelelője, írás (BL) szerint.

A 02,03,04 és 05-ös értékekkel tetszőleges vezérlő karaktersorozatok küldhetők vagy fogadhatók.

AL = 06 az inputra, és

AL = 07 az outputra való felkészültség állapotát kérdezi le, a következő módon:

Ha file-számot adunk meg a 06 és 07-es értékekhez, akkor AL-ben mindaddig FF-et kapunk vissza, míg a file végét el nem érjük, ezután pedig 00-t, hacsak át nem állítjuk a 42-es funkcióval az írási/olvasási mutatót.

Ha perifériát adunk meg, FF-et kapunk AL-ben készenléti, és 00-t nem készenléti állapot esetén.

AL = 08 azt kérdezi meg, hogy a BL-ben megadott egység cserélhető vagy sem. Ha igen, akkor a visszatéréskor AX = 0, különben 1.

AL = 09 ellenőrzi, hogy a BL-ben megadott egység lokális, vagy hálózaton implementált. Ha az egység lokális, akkor a rutin DX-ben az egység perifériakezelőjének állapotszavát adja vissza. Ha nem, akkor a DX-nek a 12. bitje magas, a többi 0.

AL = 0A ellenőrzi, hogy a BX-ben megadott file-számot kezelő periféria handler hálózaton implementált vagy lokális. Ez utóbbi esetben a visszatéréskor a DX-ben az IOCTL adatok találhatók. Ha a file-szám nem lokális, akkor a DX 15.bitjét a rutin magasra állítja.

AL = 0B alfunkcióval beállíthatjuk, hogy a DOS hányszor próbálkozzon egy lemezművelettel, ha az file zárolása miatt nem hajtható végre. DX-ben kell megadni a probálkozások számát. Ha a visszatéréskor a Carry magas (= 1), akkor a file megosztási környezetet nem töltöttük be a SHARE DOS paranccsal, s ezért az alfunkció nem hajtható végre. Ilyenkor AX értéke 1.

AL = 0C segítségével a nyomtatás-míg-foglalt iterációszáma kérdezhető le. BX-ben kell a file-számot megadni. A CH értékét 05H-ra kell beállítani. Ha CL = 45H, akkor az alfunkció beállítja, ha CL = 65H, akkor lekérdezi a számlálót. A meghíváskor a DS:DX egy szóra mutat, amelyik a számlálót tartalmazza. CL értékétől függően ezt is fel kell tölteni, vagy visszatéréskor innen olvasható ki a kért információ.

AL = 0D segítségével a blokkos berendezések bizonyos adatai is beállíthatók. Megint BL-ben kell megadni az egységet. A CH értéke kötelezően 08H. A CL értéke határozza meg az elvégzendő feladatot:

- 40 = paraméterek beállítása;
- 60 = paraméterek lekérdezése;
- 41 = sáv írása a logikai egységen;
- 61 = sáv olvasása a logikai egységről;
- 42 = sáv formázása a logikai egységen;
- 62 = sáv ellenőrzése a logikai egységen;

DS:DX minden esetben az adott funkcióhoz tartozó puffer elé mutat (= puffer kezdőcíme - 1). Ezt a CL értéktől függően nekünk kell állítani, vagy onnan kapunk információt.

AL = 0E lekérdezi a logikai-fizikai egységmegfeleltetést. BX tartalmazza az egység betűjelét (0 = aktuális, 1 = A stb.) A hibát a Carry magasra állítása (= 1) jelzi. Ha nem, akkor AL annak a logikai egységnek a betűjele, amelyik utoljára használta a szóbanforgó fizikai egységet. Ez 0, ha a fizikai egységhez csak egy logikai egység tartozik.

AL = 0F beállítja a logikai-fizikai egységmegfeleltetést. BX tartalmazza az egység betűjelét (0 = aktuális, 1 = A stb.). Az alfunkció hatására a fizikai egységet a soronkövetkező logikai egységhez rendeli hozzá a rendszer. A hibát a Carry magasra állítása (= 1) jelzi. Ha nem, akkor AL annak a logikai egységnek a betűjele, amelyik újonnan használja a fizikai egységet.

Megjegyzés:

Valamely file-szám akkor kezel perifériát, ha a megnyitáskor (3D funkció) megadott filenév a DOS számára periféria-név (pl. standard PRN), vagy a CONFIG.SYS-ben kijelölt valamelyik felhasználói periféria-kezelő rutin tartalmazza a nevet.

A 44-es funkció outputja:

A 02,03,04 és 05-ös (AL) esetén az átvitt byte-ok számát, a 06 és 07 esetén a készenléti vagy nem készenléti állapotot kapjuk hibátlan működés esetén AX-ben.

Hiba esetén a Carry magas lesz(= 1).

Az AL = 0 esetén a funkciónak DX-ben is van outputja.

Ha AL = 0 vagy 1, DX bitjei a következő információt tartalmazzák:

```
   15        14    13...8      7 6 5     4          3 2 1 0
fenntartott  C   fenntartott   I E B  fenntartott   I I I I
             T                 S O I               S S S S
             R                 D F N               C N C C
             L                 E                   L U O I
                               V                   K L T N
```

ahol:

ISDEV = 1 karakteres berendezés esetén.
= 0 blokkos berendezés (lemez-file) esetén (ekkor a 8-15 bitek 0 értékűek).

ISDEV = 1 esetén:

- EOF = 0, ha input esetén EOF-rekordot olvasott,
- BIN = 1, ha bináris módon történik a feldolgozás (a CTRL-Z-t nem ellenőrzi),
  = 0, ha ASCII módon történik (a CTRL-Z-t EOF-ként értelmezi),
- ISCLK = 1, ha a berendezés óra (clock device),
- ISNUL = 1, ha ez null-egység (null device).

A null-egység egy nem létező berendezés. Használhatjuk DOS parancsban is, de nem küld és nem fogad információt. Például, ha egy COPY parancsnak ez az outputja, a COPY a forrásról olvas, de nem ír sehova - ezt a lehetőséget ellenőrzésre lehet használni.

- ISCOT = 1, ha ez a periféria a konzol output.
- ISCIN = 1, ha ez a periféria a konzol input.
- CTRL = 0, ha az AL = 2,3,4,5 értékek erre a perifériára nem alkalmazhatók.
- CTRL = 1, ha a 2,3 műveleti kód is kiadható. Ezt a bitet beállítani nem lehet a funkcióval.

ISDEV = 0 esetén:

EOF = 0, ha az adatbevitel befejeződött.

A 0-5 bitek blokkos periféria esetén a periféria azonosítóját (0 = A, 1 = B stb.) adják meg.

A 15, 8-13 és 4-es bitek értékét a felhasználó nem módosíthatja.

17.3.67 45-ös funkció - egy file file-számának duplikálása

A funkció BX-ben egy nyitott file file-számát várja.

AX-ben visszaad egy másik file-számot, amely ugyanarra a file-ra és ugyanarra a pozícióra (mutató-állásra) mutat, mint az eredeti file. Az új file attributum byte-ja ugyanaz, mint a régié. A továbbiakban ugyanannak a file-nak két file-száma lesz.

Példa:

```
procedure DuplicateHandle(handle: integer;
                          var handle2: integer;
                          mode: integer);
begin
   reg.BX:=handle;
   reg.AH:=$45;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else begin
     mode:=0;
     handle2:=reg.AX
   end
end;
```

17.3.68 46-os funkció - file-szám duplikálás úgy, hogy a másodpéldányt a felhasználó adja meg

A funkció BX-ben egy létező file file-számot, CX-ben egy másik számot vár.

Ha CX egy nyitott file-számot tartalmaz, ezt a file-t a funkció lezárja, majd a CX-beli file-számot átirányítja úgy, hogy az ugyanarra a file-ra és annak ugyanarra a pozíciójára mutasson a további file-kezdeti utasítások szempontjából, mint a BX-beli file-szám.

Ezután, ha a két file-szám bármelyikét felhasználva, a file írási/olvasási mutatóját átállítjuk, a másik file-szám pointere ugyanúgy változik.

Példa:

```
procedure ForceDuplicateHandle(handle: integer;
                          handle2: integer;
                          mode: integer);
begin
   reg.BX:=handle;
   reg.CX:=handle2;
```

```
   reg.AH:=$46;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else mode:=0;
   handle2:=reg.AX;
end;
```

17.3.69 47-es funkció - adott meghajtó aktuális alkönyvtárának lekérdezése

A funkció DS:SI-ben egy 64 byte-os felhasználói memóriaterület címét , DL-ben a meghajtó azonosítóját (0=aktuális, 1=A stb.) várja. Erre a memóriaterületre teszi a gyökér tartalomjegyzéktől részletesen kifejtett elérési utat. (A meghajtó betűazonosítója ebben nem szerepel). Az utat leíró karaktersor nem kezdődik backslash-sel(\). A string végét egy 0 byte jelzi.

Példa:

```
procedure GetDirectory(var path:string64
                           drive: byte;
                           var mode:integer);
begin
   reg.DS:=Seg(path);
   reg.DX:=Ofs(path);
   reg.DL:=drive;
   reg.AH:=$47;
   MSDos(reg);
   if (reg.Flags and 1)=0 then mode:=0
   else mode:=reg.AX
end;
```

17.3.70 48-as funkció - memóriafoglalás

A funkció BX-ben várja a kívánt memóriméretet, paragrafusban kifejezve. (Lásd: a DOS munkaterületei című fejezet Memória nyilvántartás című részét.)

A funkcióból való visszatéréskor AX:0 mutat a lefoglalt memória blokkra. Ha a kért mennyiség nem áll rendelkezésre, a memóriából még lefoglalható mennyiséget BX-ben kapjuk paragrafusokban kifejezve.

Példa:

```
procedure AllocateMemory(var memory: integer;
                           var address: integer;
                           var mode: integer);
begin
   reg.BX:=memory;
```

```
   reg.AH:=$48;
   MSDos(reg);
   if (reg.Flags and 1)=1 then begin
     mode:=AX;
     memory:=reg.BX
   end
     else begin
     mode:=0;
     address:=reg.AX
   end
end;
```

17.3.71 49-es funkció - lefoglalt memóri felszabadítása

A funkciónak ES-ben adjuk meg a visszaadandó memóriablokk szegmenscímét. (A memóriablokkok szegmenshatáron kezdődnek.) Ezt a funkció a rendszernek továbbítja, azaz ez a memóriablokk felszabadul. (A DOS minden memóriablokk elején felépít egy vezérlő blokkot és ebben tartja nyilván, hogy az illető blokk, illetve annak egy része foglalt-e vagy szabad.)

Példa:

```
procedure FreeMemory(page: integer;
                     var mode: integer);
begin
   reg.ES:=page;
   reg.AH:=$49;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
      else mode:=0;
   handle2:=reg.AX
end;
```

17.3.72 4A funkció - lefoglalt memóriablokk méretének megváltoztatása

A funkciónak ES-ben adjuk meg a módosítandó memóriablokk szegmenscímét, BX-ben a kívánt új blokkméretet paragrafusban kifejezve.

A funkció hívásakor a DOS megpróbálja a megadott blokk méretét a megadott méretre növelni vagy csökkenteni. Ha a méret növelése nem sikerül, BX-ben adja vissza a rendelkezésre álló maximális blokkméretet. Ilyenkor Carry-t a rendszer magasra állítja (=1).

Példa:

```
procedure ModifyMemory(page: integer;
                       var size: integer;
```

```
                                var mode: integer);
begin
   reg.BX:=size;
   reg.ES:=page;
   reg.AH:=$4A;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else begin
     mode:=0;
     size:=reg.BX
   end
end;
```

17.3.73 4B funkció - program betöltése végrehajtatással vagy anélkül

A funkció lehetővé teszi, hogy a felhasználói program egy másik programot betöltsön, akár végre is hajtasson.

DS:DX-ben a funkció egy olyan ASCII sztring címét várja, amely tartalmazza a betöltendő program file-specifikációját. ES:BX a betöltéséhez szükséges paraméter-blokkra mutat. A paraméter-blokk tartalmát AL értéke szabja meg. Hasonlóan az AL értéke határozza meg, hogy a DOS elindítja-e a betöltött programot:

AL = 0 esetben a DOS létrehozza a program számára a PSP-t, a befejezés és a Ctrl-Break esetén felhasználandó kilépési címeket pedig a hívó programba irányítja a funkcióhívás utáni helyre, ezután pedig elindítja a betöltött programot.

AL = 3 esetében a DOS nem hoz létre PSP-t, és nem indítja el, csak betölti a programot.

Megjegyzés:

AL = 0 esetén, amikor a főprogram visszakapja a vezérlést, az összes regiszter és a verem (stack) tartalma is változik, így az SS, SP és a többi szükséges regisztert a funkció hívása előtt ki kell menteni. Amikor a hívott program befejezte a működését, vissza lehet tölteni.

A paraméter-blokk tartalma a következő:

a) *AL = 00*

1. szó A betöltendő program programkörnyezetét megadó (legfeljebb 32K hosszú) sztring szegmenscíme. Ha ez 0, akkor a betöltendő program örökli a hívó program programkörnyezetét. A környezet szegmenscíme, amint az a PSP leírásában is látható, a betöltendő program PSP-jének 2C helyére (PSP + 2CH) kerül.

A fent említett sztringben megadható lehetőségeket szintén a PSP leírásánál ismertetjük. A környezetleíró sztring paragrafushatáron kezdődik.

2.-3.szó Arra a parancsra mutat, amelyet a rendszer a betöltendő program PSP + 80 helyére tesz.

4.-5.szó A betöltendő program PSP + 5C helyére teendő FCB-re mutat.

6.-7.szó A PSP + 6C-re teendő FCB-re mutat.

b) *AL = 03*

1. szó Az a szegmenscím, ahová a file-t be lehet tölteni.

2. szó Az az eltolási(relocation) érték, aminek figyelembe vételével a programot be kell tölteni (lásd az .EXE file szerkezetének ismertetését).

Megjegyzés:

Az összes olyan file, amelyet a főprogram megnyitott, a betöltendő program számára is nyitva van, elérhető. Így egyes részfeldolgozási feladatokat kényelmesen struktúrálhatunk.

A 4B funkció a betöltéshez a COMMAND.COM-ban lévő betöltő programot használja. Ha a felhasználói program a betöltő programot átfedte, akkor ezzel a funkcióhívással újból a memóriába kerül. Ha a felhasználó a teljes memóriaterületet igényelte, akkor a funkcióhívás hibaüzenetet ad, mivel nincs hely a betöltő program új példánya számára.

17.3.74 4C funkció - program befejezése

A funkció hívásakor - opcionálisan - AL-be egy bináris számot, úgynevezett befejezési kódot tehetünk. A funkció az éppen futó programot befejezi, azokat a file-okat, amelyeket megnyitott, lezárja és a vezérlést a parancsfeldolgozónak, illetve a hívó programnak adja át. A befejezési kódot az IF alparancs IF ERRORLEVEL formájával, vagy a 4D funkcióval le lehet kérdezni. Más outputja nincs a funkciónak.

Példa (a 4B és 4C funkciókhoz):

```
procedure LoadProgram (ProgramName:szoveg;
                       ParameterBlock:szoveg;
                       OverlayFlag:boolean;
                       var mode: integer);
begin
   reg.DS:=Seg(ProgramName);
   reg.DX:=Ofs(ProgramName);
   reg.ES:=Seg(ParameterBlock);
   reg.BX:=Ofs(ParameterBlock);
   if overlayFlag then reg.AL=$03 else reg.AL=0;
   reg.AH:=$4B;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else mode:0
end;

procedure EndProcess(ReturnCode: byte);
begin
   reg.AL:=ReturnCode;
```

```
   reg.AH:=$4C;
   MSDos(reg)
end;

function GetReturnCode:integer;
begin
   reg.AH=$4D;
   MSDos(reg);
   GetReturnCode:=reg.AX
end;
```

17.3.75 4D funkció - a hívott program befejezési kódjának lekérdezése

A hívó program ezzel a funkcióval kaphatja meg - az AL regiszterben - azt a befejezési kódot, amelyet a 31-es vagy 4C funkcióval a hívott program üzen.

AH értéke a következő lehet:

- 00 = normál befejezés;
- 01 = CTRL-BREAK befejezés;
- 02 = befejezés periféria hiba miatt;
- 03 = befejezés 31-es funkcióval

Megjegyzés:

Amikor a kódot lekérdezzük, dolgozzuk is azonnal fel, mert a 4D funkció csak egyszer adja vissza. Az ismételt hívás eredménytelen lesz.

17.3.76 4E funkció - elérési úttal megadott file keresése az első találatig

A funkció DS:DX-ben egy olyan ASCII sztring címét várja, amely a keresett file meghajtóját, elérési útját és nevét tartalmazza (szerepelhetnek globális file-név karakterek is). CX-ben az attributum byte értékét kell megadni, ami megmutatja, hogy milyen file-okat keresünk.

A funkció a DTA-ba adja vissza az először megtalált egyező file információit a következő módon:

21 byte	fenntartva a 4E funkciót esetleg követő 4F funkcióhívás számára;
1 byte	a megtalált file attributuma;
2 byte	létrehozásának ideje;
2 byte	létrehozásának dátuma;
2 byte	méretének alacsonyabb helyiértékű szava;
2 byte	méretének magasabb helyiértékű szava;
12 byte	a file neve és kiterjesztése (szóközök nélkül és az esetleges kiterjesztés előtt pont van, azaz olyan formátumban, ahogy egy parancsban paraméter lehet),
1 byte	0 tartalmú.

Példa:

```
procedure FindFirstFile (FileName: szoveg;
                         Attributes: integer;
                         var mode: integer);
begin
   reg.DS:=Seg(FileName);
   reg.DX:=Ofs(FileName);
   reg.CX:=Attributes;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else mode:=0;
     handle2:=reg.AX
   end
end;
```

17.3.77 4F funkció - további, a megadottnak megfelelő file keresése (4E hívás után)

A funkció inputja ugyanaz, mint a 4F funkcióé, feladata pedig a 4E hívása után további megfelelő file-ok keresése. Ha a 4F további megfelelő file-okat talál, a DTA-t úgy tölti fel, mint 4E-nél, egyébként hibakódot ad. Ez akkor is megtörténik, ha nem talál megfelelő file-t.

Példa:

```
procedure FindNextFile (mode: integer);
begin
   reg.AH:=$4F;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else mode:=0
end;
```

17.3.78 50-53-as funkció - DOS számára fenntartott funkció

17.3.79 54-es funkció - VERIFY kapcsoló értékének lekérdezése

A VERIFY kapcsoló értéke határozza meg, hogy végezzen-e a rendszer ellenőrzést írás után (lásd az ugyanilyen nevű DOS parancsot).

Az 54-es funkció segítségével hozhatjuk be ezt az értéket programunkba.

A funkció inputot nem kíván.

Végrehajtása után AL értéke adja a kapcsoló értékét:

AL = 00, ha az érték OFF és
AL = 01, ha ON.

Megjegyzés:

A 2E funkcióval állíthatjuk be a kapcsoló értékét.

Példa:

```
function GetVerifyState: byte;
begin
   reg.AH:=$54;
   MSDos(reg);
   GetVerifyState:= reg.AL
end;
```

17.3.80 55-ös funkció - DOS számára fenntartott funkció

17.3.81 56-os funkció - file átnevezése

A funkció DS:DX-ben egy komplett file-specifikációt tartalmaz. A névben nem szerepelhetnek globális file-név karakterek. ES:DI-ben pedig azon ASCII string címét kell megadni, amely az új nevet a következőképpen adja meg: Ha szerepel meghajtó, az ugyanaz, mint az első string-ben. Az elérési út és a file-név természetesen más is lehet. A file-név a második string-ben sem tartalmazhat globális file-név karaktereket.

Példa:

```
procedure RenameFile (FileNameOld,FileNameNew:szoveg;
                      var mode: integer);
begin
   reg.DS:=Seg(FileNameOld);
   reg.DX:=Ofs(FileNameOld);
   reg.ES:=Seg(FileNameNew);
   reg.DI:=Ofs(FileNameNew);
   reg.AH:=$56;
   MSDos(reg);
if (regFlags and 1)=1 then mode:=reg.AX
   else mode:=0
end;
```

17.3.82 57-es funkció - file dátumának és időpontjának lekérdezése, illetve módosítása

Az, hogy a funkció lekérdezést vagy beállítást végez-e, AL tartalmától függ:

AL = 00 lekérdezés,
AL = 01 beállítás.

BX-ben adjuk meg a file-számot és AL = 01 esetén CX-ben a beállítandó időpontot, DX-ben pedig a beállítandó dátumot. Ez utóbbi két adatot ugyanabban az alakban kell megadni, ahogy ezek a tartalomjegyzék bejegyzésben szerepelnek, csak a byte-ok sorrendjét az egyes regiszterekben meg kell fordítani, azaz pl. DH-ban kell a dátum alacsonyabb helyiértékű részét megadni.

Ha a funkció feladata a beállítás volt, az esetleges hibakódon kívül nincs outputja. Ha lekérdezés volt, akkor CX-ben kapjuk a file létrehozásának, illetve utolsó módosításának időpontját, DX-ben a dátumát.

17.3.83 58-as funkció - memória helyfoglalás módjának lekérdezése, illetve beállítása

A funkció inputjaként AL-ben 0-t kell megadni, ha lekérdezni, 1-et, ha beállítani akarjuk a memória helyfoglalás módját. Ez utóbbi esetben BX-ben a választott helyfoglalás kódját kell megadni. Lekérdezéskor ezt a kódot AX-ben kapjuk vissza.

A funkció csak a DOS 3.20-től használható!

A lehetséges kódok és jelentésük:

0 **Első találat.** A DOS az első - a kérésnek megfelelő memóriablokkot foglalja le. Ez a kezdeti (default) mód is.
1 **Legjobb találat.** A DOS végigvizsgálja az összes lehetséges memóriablokkot, s a kérésnek megfelelők közül a legkisebbet foglalja le.
2 **Utolsó találat.** A DOS a legmagasabb helyen található blokkoktól lefelé kezd el keresni, s az első megfelelő blokkot foglalja le.

Ha hiba történik (AL vagy BX tartalma nem megfelelő), akkor a DOS a Carry flag-et magasra (= 1) állítja.

Példa:

```
procedure AllocationStrategy(code: byte;
                             var method: integer;
                             var mode: integer);
begin
   reg.AH:=$58;
   if code=1 then reg.BX:=method;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
```

```
   else
   if code=0 then method:=reg.AX
end;
```

17.3.84 59-es funkció - a hiba leírásának lekérdezése

Ennek a funkciónak a segítségével a hiba okáról további információkat kaphatunk. A felhasználó által készitett 24H megszakítás kezelők pl. használhatják ezt a funkciót a hibák elemzésére. A funkció meghívásakor a BX-be 0-t kell helyezni, ez a rendszer továbbfejlesztésekor majd jelzi, hogy milyen verziójú hibakezelő rendszert használunk.

A funkció csak a DOS 3.00-tól használható!

A funkció outputja:

BH tartalmazza a hiba osztályát. Az egyes értékek jelentése:

1 = Nem megengedett paraméter (pl. memória, csatorna-szám)
2 = Ideiglenes probléma (pl. file zárolása)
3 = Hozzáférési probléma
4 = Software rendszer belső hibája
5 = Hardware hiba
6 = Software rendszer hiba, amit nem a processz maga okozott (pl. nem megfelelő CONFIG.SYS file)
7 = Alkalmazói program hiba
8 = A file vagy valami egyéb nem található
9 = A file vagy valami egyéb nem megfelelő formátumú
10 = Zárolt file vagy valami egyéb tétel
11 = Rossz lemez vagy egyéb tárolási eszköz probléma
12 = Egyéb hiba
13 = Ismeretlen hiba

BL tartalmazza a javasolt hibajavítási mód kódját. Az egyes kódok jelentése a következő:

1 = Próbálkozzon újra, majd kérdezze meg a felhasználót.
2 = Próbálkozzon egy kis szünet után.
3 = Ha a felhasználó által megadott adat volt, kérdezze meg újra.
4 = Fejezze be a karbantartási funkciók elvégzése után.
5 = Azonnal be kell fejezni!
6 = A hiba csak informális jellegű, nincs igazi hiba.
7 = Kérdezze meg a felhasználót néhány konkrét dologról, majd ismételje meg (pl. cserélje ki a lemezt).

A CH **a hiba helyére utaló** információt tartalmaz:

1 = ismeretlen;
2 = véletlen elérésű eszköz (pl. lemezegység);

3 = hálózat;
4 = soros elérésű eszköz;
5 = a memória.

Példa:

```
procedure GetError(var ErrorCode:integer;
                   var ErrorClass,Action,Locus: byte);
begin
   reg.AH:=$59;
   reg.BX:=0;
   MSDos(reg);
   ErrorCode:=reg.AX;
   ErrorClass:=reg.BH;
   Action:=reg.BL;
   Locus:=reg.CL;
end;
```

17.3.85 5A funkció - file létrehozása egyedi névvel

A funkció egy olyan file-t hoz létre, amilyen nevű file még nincs a rendszerben. A funkció meghívásakor DS:DX egy út nevet leíró sztringre mutat, amelyik 0 byte-tal végződik. Ennek az útnak megfelelő szótárban fog a funkció egy új file-t létrehozni. CX a létrehozandó file attributumait tartalmazza.

A funkció csak a DOS 3.00-tól használható!

A visszatéréskor AX a megnyitott file-hoz rendelt file-számot tartalmazza, míg a funkció a DS:DX sztringhez hozzáírja a file nevét. Így az egyéb műveletekben használható.

A hívó processzor törlése nem törli az ideiglenes file-t, arról magunknak kell gondoskodni.

Ha hiba történik, akkor a Carry flag magas (= 1) lesz, s AX tartalmazza a hiba kódját:

2 = A file attribútum nem megfelelő
3 = A megadott útleírás nem megfelelő
4 = Túl sok a megnyitott file
5 = Az elérés nem megengedett

Példa:

```
procedure CreateUniqueFile (var PathName:szoveg;
                            Attributes: integer;
                            var handle: integer;
                            var mode: integer);
begin
   reg.DS:=Seg(PathName);
   reg.DX:=Ofs(PathName);
```

```
   reg.CX:=Attributes;
   reg.AH:=$5A;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else begin
     mode:=0;
     handle:=reg.AX
   end
end;
```

17.3.86 5B funkció - új file létrehozása

A funkció segítségével egy még nem létező file-t hozhatunk létre. A meghíváskor DS:DX a file nevére mutat (végén egy 0 byte-tal), és CX a létrehozandó file attribútumait tartalmazza. Ha a file még nem létezik, akkor a funkció létrehozza, megnyitja kompatibilis módban, s egy file számot rendel hozzá. Ha létezik - akkor szemben a 3C funkcióval - nem csonkolja 0 hosszúságra, hanem hibajelzést ad.

Visszatéréskor csak hibaüzenetet kapunk. A Carry flag magas (=1) volta jelzi a hibát. AX a hiba okának kódját tartalmazza:

2 = a file-megadás hibás
3 = a megadott útleírás nem megfelelő
4 = az elérés nem megengedett
80 = a file már létezik

Megjegyzés:

A funkció kiválóan alkalmas szemafornak (test and set) hálózaton együttműködő programoknál.

Példa:

```
procedure CreateNewFile (var PathName:szoveg;
                         Attributes: integer;
                         var Handle: integer;
                         var Error: boolean);
begin
   reg.DS:=Seg(PathName);
   reg.DX:=Ofs(PathName);
   reg.CX:=Attributes;
   reg.AH:=$5B;
   MSDos(reg);
   if (reg.Flags and 1)=1 then Error:=true
   else begin
     Error:=false;
     Handle:=reg.AX
   end
end;
```

17.3.87 5C funkció - file-zárolás

A funkció két alfunkcióra tagolódik: 00 = zárolás és 01 = zárolás feloldása. A zárolással megtiltható, hogy a megadott file kijelölt területéhez más program hozzáférjen. A feloldással a kijelölt területet ismét elérhetővé tehetjük más programok számára is. E funkciót kell használni hálózati alkalmazásoknál, ha ugyanazon file-okon egyidőben több munkahelyről is akarnak dolgozni. A funkciókódot (00 vagy 01) az AL regiszterben, a file-számot a BX-ben, a zárolni kívánt file-terület kezdetét - a byte helyét a file elejéhez viszonyítva - a CX:DX regiszterpárban, a zárolandó szakasz hosszát - byte-ban - az SI:DI regiszterpárban kell megadni.

Visszatéréskor csak hibaüzenetet kapunk. A Carry flag magas (=1) volta jelzi a hibát. AX a hiba okának kódját tartalmazza:

1 = érvénytelen alfunkciókód
6 = érvénytelen file-szám
33 = zárolási problémát
36 = a megosztást tároló puffer túlcsordult

Figyelem! Ha egy olyan file-t zárunk le, amelyben zárolt részek vannak, akkor a file tönkremehet! Lezárás előtt ezért valamennyi zárolást oldjunk fel, s csak utána zárjuk le a file-t.

Példa:

```
procedure Locking(Code: byte;handle: integer;
                  FirstByte, LengthOfRegion: integer2;
                  var mode: integer);
begin
   reg.AL:=Code;
   reg.BX:=handle;
   reg.CX:=FirstByte.High;
   reg.DX:=FirstByte.Low;
   reg.SI:=LengthOfRegion.High;
   reg.DI:=LengthOfRegion.Low;
   reg.AH:=$5C;
   MSDos(reg);
   if (reg.Flags and 1)=1 then mode:=reg.AX
   else mode:=0
end;
```

17.3.88 5E funkció - hálózati funkciók

Az 5E funkciók a hálózatkezeléssel kapcsolatos DOS funkciók. Az egyes alfunkciók - aminek kódját AL-ba kell elhelyezni - jelentése teljesen eltér, ezért külön ismertetjük ezeket.

A funkció csak a DOS 3.10-től használható!

00-ás alfunkció - a lokális gép nevének lekérdezése

Az alfunkció a lokális gép nevét kérdezi le. Meghívásakor a DS:DX-nek egy 16 byte-os pufferre kell mutatnia, ahová a DOS majd a gép nevét helyezi. CX a visszatéréskor a gép azonosítási számát tartalmazza. Hiba esetén (ha nincs bekapcsolva a hálózat) a Carry magas lesz (= 1) és AX-be 1-et ír a DOS.

02-es alfunkció - nyomtató inicializálása

Az alfunkció beállítja a hálózaton installált nyomtatóra - minden file előtt - elküldendő karaktersorozatot. A meghíváskor BX a hozzárendelési számot (lásd később!), CX a karaktersorozat hosszát, DS:SI pedig az első byte címét tartalmazza.

Ha hiba történt (a hálózat nem üzemel), akkor a Carry flag magas lesz (= 1) és AX-be 1 kerül.

17.3.89 5F funkció - hálózati hozzárendelések beállítása/lekérdezése

Az 5F funkció segítségével a hálózati átirányításokat lehet lekérdezni vagy beállítani. Az AL regiszterben lehet megadni az alfunkció számát, ami 02H, 03H vagy 04H lehet.

A funkció csak a DOS 3.10-es változatától használható!

02H alfunkció - hozzárendelés lekérdezése

Az alfunkció meghívásakor BX a hozzárendelési indexet kell, hogy tartalmazza (0 a lista első eleme), míg DS:SI és ES:DI egy-egy puffer kezdőcíme, ahová majd az alfunkció a lokális, illetve a távoli gép nevét helyezi el. Lehet, hogy a lokális név üres, s egyedül a terminátor 0 byte-ból áll.

Visszatéréskor az alfunkció a nevek beírásán túl, CX-ben adja vissza az 5F03-mal beállított felhasználói kódot. BL értéke 3, ha az átirányított eszköz nyomtató; 4, ha lemezegység.

A hibát a magas (= 1) Carry flag jelzi. Ekkor AX a hiba kódját tartalmazza:

1 = Érvénytelen alfunkció kód
18 = Nincs több file

03H alfunkció - hozzárendelés beállítása

Az alfunkció segítségével lehetőség van nyomtató vagy lemezegység hálózatra történő átirányítására. BL-nek a meghíváskor 3-at kell tartalmaznia, ha nyomtatóról, 4-et, ha lemezegységről van szó. CX-ben adhatjuk meg a felhasználói kódot. DS:SI egy 0 byte-tal lezárt sztringet tartalmaz, ami vagy a nyomtató neve, vagy egy lemezegység betűazonosítója kettősponttal együtt, vagy egy üres string. ES:DI a hálózaton implementált egység altartalomjegyzék nevét tartalmazza. Ennek alakja a következő:

\gép-név[\út] 00H jelszó 00H

Például a következő Turbo Pascal kifejezés egy alszótárt ad meg:

"\kozpont\main\program" + chr(0) + "mancie" + chr(0)

A fenti megadásban a távoli gép neve kozpont, azon a main\program nevű altartalomjegyzékbe irányítjuk át a lemezegység műveleteit.

Ha BL = 3, akkor a DS:SI által megadott név csak PRN, LPT1, LPT2 vagy LPT3 lehet. Ha BL = 4, akkor csak egy érvényes lemezegységet adhat meg, vagy üres lehet. Ez utóbbi esetben az adott jelszóval a hálózati tartalomjegyzéket kísérli meg elérni a rendszer.

Visszatéréskor a Carry magasra (=1) állítása jelzi, hogy hiba történt, ekkor AX tartalmazza a hiba kódját:

1 = a hálózat nem üzemel, vagy a paraméterek nem megfelelőek
3 = a megadott elérési út nem megfelelő
5 = a hálózati tartalomjegyzék/jelszó kombináció nem megfelelő
8 = nincs elég memória a sztringműveletek elvégzésére.

04H alfunkció - hozzárendelés törlése

A funkció segítségével törölhetünk egy, az 5F03 alfunkció segítségével létrehozott hozzárendelést. DS:SI tartalmazza az átirányított egység nevét. Ez vagy a PRN, LST1, LST2, LST3 nevek egyike, vagy egy érvényes lemezegység betűazonosítója a kettősponttal. A sztring végét egy 0 byte jelzi. Ha a string mindössze két \\ jelből áll, akkor ennek hatására megszűnik a távoli géppel való kapcsolat.

Hiba esetén a Carry magas (=1) lesz, s AX tartalmazza a hiba kódját:

1 = A hálózat nem üzemel, vagy nem átirányított egységre hivatkozunk;
15 = A nyomtató vagy lemezegység átirányítása a távoli gépen szünetel.

17.3.90 62 funkció - program-szegmens lekérdezése

A funkció visszatéréskor BX-ben a hívó processzor program-szegmens terület szegmens címét adja vissza. Hibajelzés nincs.

Példa:

```
function GetPSP: integer;
begin
   reg.AH:=$62;
   MSDos(reg);
   GetPSP:=regBX
end;
```

17.3.91 65 funkció - kibővített ország-specifikus információk lekérdezése

A DOS 3.30-as verziójától kezdve lényegesen bővült az ország-specifikus információk köre, pl. a kódlapok használatával. A bővebb adatok lekérdezésére egy új DOS funkciót kellett beilleszteni a rendszerbe.

A funkció meghívása előtt BX-be a kódlap sorszámát, DX-be pedig az ország kódját kell elhelyezni. Ha BX = -1, az a használt CON eszköznek felel meg. Hasonlóan, ha DX = -1, akkor a rendszer alapértelmezésében szereplő ország-kóddal dolgozik a funkció. ES:DI a memóriában levő pufferre mutat, CX a puffer hosszát adja. Ennek legalább 5-nek kell lennie. AL-be annak megfelelően az 1,2,4 vagy 6 értékek egyikét kell beírnunk attól függően, hogy milyen adatot kérünk. Az egyes értékek jelentése a következő:

AL	Jelentés
1	ország-specifikus információ
2	nagybetű konvenció
4	nagybetű konvenció file-nevekben
6	rendezési tábla

AL értékétől függően a pufferben más és más információt kapunk vissza.

Ország-specifikus információ (AL = 1)

1 byte	01
2 byte	a puffer mérete
2 byte	ország-kód
2 byte	kódlap sorszáma
2 byte	dátum formátuma
5 byte	pénznem jele
2 byte	ezresek elválasztó jele
2 byte	tizedes elválasztó jel
2 byte	dátum elválasztó jel
2 byte	idő elválasztó jel
1 byte	jelző bitek
1 byte	pénzegység használatakor a tizedes jegyek száma
1 byte	idő formátuma
2 duplaszó	a kisbetűk/nagybetűk átalakító rutin kezdőcíme
2 byte	adatelválasztó jel
10 byte	fenntartott (a DOS használja)

Nagybetű konverzió (AL = 2)

1 byte	02
1 duplaszó	a nagybetű tábla elejére

Nagybetű konverzió file-nevekre (AL = 4)

1 byte	04
1 duplaszó	a nagybetű tábla elejére

Mindkét esetben a tábla 130 byte-ból áll. Az első két byte a táblázat méretét adja meg. A táblázat n-ik bejegyzése az n + 128 ASCII karakterhez tartozó nagybetűt adja meg.

Rendezési tábla (AL = 6)
1 byte 06
1 duplaszó a rendezési tábla elejére

A rendezési tábla 258 byte-ból áll. Az első két byte a tábla hosszát tartalmazza, majd 256 ASCII karakter következik a megfelelő (tehát ABC és nem kód-szerinti) sorrendben.

17.3.92 66 funkció - globális kódlap beállítása/lekérdezése

A funkció segítségével a rendszer által használt kódlapot állíthatjuk be, vagy kérdezhetjük le. Lekérdezéskor AL = 1. Visszatéréskor a Carry-flag magas, ha hiba történt. BX tartalmazza az aktuális kódlapot, míg DX a rendszer indításakor használt kódlapot.

Beállításhoz az AL = 2 értéket kell választani. A rendszer az új kódlapot a COUNTRY.SYS file-ból tölti be a memóriába. Így a kódot külön nem kell megadni. Ha hiba történt, akkor visszatéréskor a Carry-flag magas lesz. *Megjegyezzük, hogy a funkció használata előtt az NLSFUNC program rezidens részét a memóriába kell tölteni.*

Csak a DOS 3.30-as változatától használható!

17.3.93 67 funkció - periféria-kezelők maximális számának megadása

A funkció meghívása előtt a használni kívánt periféria-kezelők számát a BX regiszterbe kell tölteni. Hiba esetén a Carry flag magas lesz. Ezzel a funkcióval maximum 64K-s periféria-kezelőt használhatunk. Ha 20-nál kisebb értéket írunk a BX regiszterbe, akkor azt 20-ra módosítja a rendszer. Megjegyezzük, hogy a CONFIG.SYS file-ban elhelyezett FILES = konfigurációs parancs segítségével maximum 255 periféria-kezelőt tudunk engedélyezni. A DOS részére helyet kell biztosítani, ahol tárolja a bővebb listát. Ezt a 4A funkcióval tehetjük meg.

Csak a DOS 3.30-as változatától használható!

17.3.94 68 funkció - puffer kiürítése

A funkció segítségével adott file-hoz tartozó puffereket a lemezre írhatjuk. BX-be kell elhelyezni a periféria-kezelő sorszámát.

Csak a DOS 3.30-as változatától használható!

A. függelék

A DOS üzenetei

Ez a függelék két részre tagolódik. Az első részben a perifériákkal kapcsolatos hibaüzenetek találhatók. Azok az üzenetek, amelyekkel a DOS felhasználórendszeréhez tartozó perifériás egységeken (például: nyomtató, képernyő, lemezmeghajtó stb.) történő írás vagy olvasás folyamán fellépő hibákat jelzi.

A második részben a DOS egyéb üzenetei találhatók. Mivel a német és angol nyelvű rendszereket használják hazánkban, ezért a hibaüzeneteket mindkét formában megadjuk. Az angol és német nyelvű üzenetek után - gondolatjelek közt - megadjuk az üzenet fordítását. Ez a legtöbb esetben tartalmi, nem szó szerinti fordítás. Ezután közöljük a hiba okát és a kijavítás módját is megadjuk a magyarázat részben. Azt is közöljük, hogy melyik program, parancs küldte az üzenetet.

A DOS egyéb hibaüzenetei a DOS 3.20-as verzió hibaüzenetei. Ennél korábbi verziókban bizonyos üzenetek még nem léteztek. A hibaüzenetek egy része bizonyos korlátozásokhoz kapcsolódik (pl.a szegmensek száma nem haladhat meg egy bizonyos értéket). A különböző verziók esetén ez az érték más és más lehet.

A.1 Perifériákkal kapcsolatos hibaüzenetek

Ha a felhasználó rendszerében perifériaműveletek (írás, olvasás) során a DOS hibát érzékel, akkor a következő formában küld üzenetet:

<TYPE> ERROR READING <DEVICE>
ABORT, RETRY, IGNORE?
(ABBRECHEN, WIEDERHOLEN, IGNORIEREN?)

vagy

<TYPE> ERROR WRITING <DEVICE>
ABORT, RETRY, IGNORE?
(ABBRECHEN, WIEDERHOLEN, IGNORIEREN?)

- <TÍPUS> HIBA AZ OLVASÁS SORÁN <EGYSÉG>
- BEFEJEZNI, ISMÉTELNI, ÁTLÉPNI?

vagy

- <TÍPUS> HIBA AZ ÍRÁS SORÁN <EGYSÉG>
- BEFEJEZNI, ISMÉTELNI, ÁTLÉPNI?

Az ilyen típusú üzenetekben az <EGYSÉG> a hibás egység nevét jelöli, mint például PRN vagy B:. A <TÍPUS> a következő hibatípusokat jelölheti:

BAD CALL FORMAT (FALSCHER AUFRUFFORMAT)
- hiba a hívás formájában

A perifériakezelőnek egy nem megfelelő hosszúságú parancs lett átadva. A hibát ki kell javítani, s a rendszert és a programot újraindítani.

BAD COMMAND (FALSHER BEFEHL)
- hibás parancs

A perifériakezelő az <EGYSÉG>-re hibás parancsot adott ki. A programot kijavítás után lehet csak újrafuttatni.

BAD UNIT (FALSHE EINMEIT)
- hibás meghajtó

A perifériakezelő egy nem létező al-egységre (meghajtóra) hivatkozott. A programot kijavítás után lehet csak újraindítani.

DATA (DATAN)
- adathiba

Az írás vagy olvasás során adathiba fordult elő. Ez általában azt jelenti, hogy fizikailag hibás rész van a lemezen.

DRIVE NOT READY (NICHT BEREIT)
- a lemezegység nincs készenléti állapotban

A megadott egységben:
- nincs lemez;
- nincs bekapcsolva;
- nincs becsukva az ajtó;
- nem megfelelően formázott lemez van.

FCB UNAVAILABLE
- FCB nem használható

A file megosztási környezet betöltése után, az FCBS konfigurációs parancsban megadottnál több file-t akarunk megnyitni.

GENERAL FAILURE
- ismeretlen hiba

A rendszer nem tudta közelebbről azonosítani a hibát. A legvaloszínűbb, hogy inkompatibilis, vagy nem megfelelően formázott lemezt használtunk. Próbálkozzunk először az ismétléssel (RETRY, WIEDERHOLENE)

NO PAPER (PAPIERENDE)
- a papír elfogyott

A nyomtatóból kifogyott a papír, nincs bekapcsolva, vagy OFF-LINE üzemmódban van.

NON-DOS DISK (KEINE DOS DISKETTE)
- nem DOS lemez

A file-leíró táblázat (FAT) hibát tartalmaz. Újra kell a lemezt formázni.

NOT READY (NICHT BEREIT)
- nem kész

Ha az üzenet a nyomtatóra vonatkozik, akkor az még nyomtat. Ha a lemezegységre, akkor annak ajtaja nyitva van, vagy nem formázott lemez van benne.

READ FAULT (LESEFEHLER)
- olvasási hiba

A DOS nem tudta sikeresen elolvasni az adatokat.

SECTOR NOT FOUND (SECTORE NICHT GEFUNDEN)
- a szektor nem található meg

Az adatokat tartalmazó szektort nem lehetett a lemezen lokalizálni. Ez akkor történhet, ha a lemezen hibás területek vannak.

SEEK (SUCHEN)
- keresési hiba

A fix vagy hajlékony lemez nem tudja a megadott sávot megtalálni.

SHARING VIOLATON
- megosztási hiba

SHARE. A file-t olyankor kíséreltük megnyitni, amikor az zárolt volt. Először kísérletezzünk az ismétléssel.

WRITE FAULT (SCHEIBFEHLER)
- íráshiba

Az adatokat nem tudta a DOS az adathordozóra írni.

WRITE PROTECT (SCHREIBSCHUTZ)
- írásvédelmi hiba

Írásvédett lemezre akartunk írni.

Megjegyzés:

Ha az üzenetek közül valamelyik a lemezmeghajtóra vonatkozik, akkor a lemezt nem szabad lecserélni. A rendszer addig várakozási állapotba kerül, amíg a felhasználó a következő válaszok valamelyikét meg nem adja:

A/A - befejezés. A rendszer befejezi azt a programot, amely a műveletet végezte.

R/W - ismétlés. A rendszer újra megismétli a műveletet. Érdemes először ezzel próbálkozni.

I/I - átlépés. A hibát átlépi, a program végrehajtását folytatja. Ezt nem javasoljuk, mivel adatok veszhetnek el.

A.2 A DOS egyéb üzenetei

A...

ABORT EDIT (Y/N)?
- Szerkesztés megszakítása (I/N)?

EDLIN. A Q(uit) parancs kiadása után visszakérdez a rendszer, hogy biztos-e, hogy abbahagyjuk a szerkesztést. Y megnyomására visszatérünk a DOS-hoz, N-re folytathatjuk a szerkesztést.

ABOUT TO GENERATE .EXE FILE CHANGE DISKS < HIT ENTER >
(BEREIT ZUM GENERIEREN DER .EXE-DATEI. DISKETTE AUSTAUSCHEN < EINGABE BETATIGEN >)
- Kész az .EXE kiterjesztésű file generálására. Cserélje ki a lemezt, < adjon ENTER-t >

LINK. Ez az üzenet akkor jelenik meg, ha a /PAUSE paramétert megadjuk. Ebben az esetben az adatállományt tartalmazó lemezt a meghajtóba kell helyezni, és le kell nyomni az ENTER billentyűt.

ACCESS DENIED
- Az elérés tilos

Az üzenetet akármelyik parancs küldheti. Valamelyik file-t, alszótárat, vagy perifériát nem megengedett módon próbáltunk használni, például csak olvasható file-ba akartunk írni.

ACTIVE CODE PAGE
- Aktív kódlap

CHCP. A DOS által használt kódlap sorszáma. Egyes periféria-kezelők ettől eltérő kódlapot használhatnak. Infofrmációs üzenet.

ACTIVE CODE PAGE FOR DEVICE DDD IS NNN
- Aktív kódlap a ddd egységen:nnn

MODE. A MODE parancs segítségével valamelyik periféria-kezelő állapotát kérdeztük le. Információs üzenet.

ACTIVE CODE PAGE NOT AVAILABLE FROM CDN DEVICE
- Nincs aktív kódlap a konzolon

KEYB. A KEYB segítségével aa CON: ,konzol állapotát kérdeztük le. A program nem

tudta a kódlap sorszámát megállapítani. Ennek oka, hogy a kódlapokhoz szükséges meghajtó nincs installálva, vagy, hogy nem töltöttünk be kódlapot.

ADD d:\út\file-név?
- Hozzáadja a d:\út\file-név file-t?

REPLACE. A /P paraméter megadásakor a parancs visszakérdez, hogy hozzáadja-e a file-t, vagy sem. Az Y az igen, az N a nem válasz.

ADDING d:\út\file-név
- d\út\file-név hozzáadva

REPLACE. A szóbanforgó file-t a rendszer az alszótárhoz adta

ALL FILES CANCELED BY OPERATOR
(ALLE DATEINEN VON BEDIENER GESTRICHEN)
- Valamennyi specifikált file nyomtatását törölte az operátor

PRINT. Ez az üzenet akkor jelenik meg, ha a felhasználó a file-ok kinyomtatását a /T paraméterrel megszakítja.

ALL LOGICAL DRIVES DELETED IN THE EXTENDED DOS PARTITION
- Valamennyi logikai egységet töröltük a kiterjesztett DOS partícióból

FDISK. A kiterjesztett DOS partíciónnincs egyetlen logikai egység sem. A partíciót célszerű törölni. Információs üzenet.

ALL SPECIFIED FILE(S) ARE CONTIGUOUS
(ALLE ANGEGEBENEN DATEINEN SIND ZUSAMMENHÄNGEND)
- Valamennyi specifikált file folyamatosan helyezkedik el

CHKDSK. A megadott file-ok egybefüggő területen vannak a lemezen elhelyezve.

ALLOCATION ERROR FOR FILE, SIZE ADJUSTED
(ZUORDNUNGSFEHLER, GRÖ E BERICHTIGT)
- File lefoglalási hiba, méret javítva

CHKDSK. Az üzenet előtt egy azonosító áll. A file-leíró táblázatban (FAT) érvénytelen szektorszám található. A file-t csonkítja (rövidíti).

AMBIGUOUS SWITCH:Z
(ZWEIDEUTIGER SCHALTER:Z)
- Kettős jelentesű kapcsoló

LINK. A Z-vel megadott jelek nem egyértelműen LINK paraméterek. Ebben az esetben a paraméter azonosítóból több jelet kell megadni.

AMOUNT READ LESS THAN SIZE IN HEADER
- A valódi méret kisebb a fejlécben megadottnál

EXE2BIN. A program file hossza kevesebb volt, mint amit a fejléc jelzett. Feltehetően újra kell szerkeszteni a programot.

APPEND ALREADY INSTALLED
- Az APPEND rezidens részét már installáltuk

APPEND. Az APPEND-et csak egyszer lehet a tárba tölteni, mi másodszor is megkíséreltük. Az APPEND-et használhatjuk.

APPEND/ASSIGN CONFLICT
- APPEND és ASSIGN helytelen együttes használata

APPEND. Az ASSIGN parancsot az APPEND előtt használtuk. Csak fordított sorrendben lehet.

APPEND/TOPVIEW CONFLICT
- APPEND és TOPVIEW helytelen együttes használata

APPEND. A TOPVIEW és az APPEND programokat rossz sorrendben indítottuk. Ki kell lépni a TOPVIEW-ból, betölteni az APPEND-et, és újból indítani a TOPVIEW-t.

AN INTERNAL FAILURE HAS OCCURED
(EIN INTERNER FEHLER IST AUFGETRETEN)
- Belső hiba a szerkesztés folyamán

LINK. A LINK működése alatt hiba jelentkezett. Ha ismétlődik, akkor a LINK programot le kell cserélni. A hibát a rendszer forgalmazójának is jelenteni kell.

ARRAY ELEMENT SIZE MISMATCH
- A tömb mérete eltér

LINK. Egy távoli (FAR) közös tömböt eltérő mérettel deklaráltunk.

ATTEMPT TO ACCESS DATA OUTSIDE OF SEGMENT SOUNDS
- Szegmenshatáron túl akartunk lépni

LINK. Hibás object file.

ATTEMPT TO PUT SEGMENT XXXXXX IN MORE THAN ONE GROUP IN FILE XXXXXX
- Az xxxxxx szegmens az xxxxxx file-ban két csoportban van.

LINK. Egy szegmenst két különböző csoportban deklaráltuk.

ATTEPMPTED WRITE - PROTECT VIOLATION
(SCHREIBSCHUTZ - SCHREIBEN NICHT MÖGLICH)
- Kísérlet történt az írásvédelem megsértésére

FORMAT. Az éppen formázás alatt álló lemez írásvédelemmel rendelkezik. A lemezt ki kell cserélni, majd akármelyik billentyű lenyomásával folytatható a formázás.

B...

**** BACKING UP FILES TO TARGET DRIVE x****
TARGET NUMBER: y
***** BACKING UP FILES TO DRIVE x *****
NUMBER: y
- Az x egységről az y file mentése megkezdődött

BACKUP. Az elmentett file sorszámát jelző üzenet. Hiba nem történt.

BACKUP FILE SEQUENCE ERROR
(DISKETTE NICHT FOLGERICHTING)
- A lemez nem megfelelő

RESTORE. Egy ismételten előállítandó file több lemezen található. A felhasználó nem azt a lemezt helyezte be, amelyen az állomány első része található. A helyes lemez betétele után a RESTORE-t újra kell indítani.

BAD COMMAND OR FILE NAME
(FALSHER BEFEHL ODER DATEINAME)
- Helytelen parancs vagy file-név

DOS. Az éppen beadott parancs érvénytelen. Adjuk ki újra, helyesen! Ha a parancs korrekt, akkor ellenőrizni kell, hogy a parancshoz tartozó file(-ok) a megadott lemezen van(nak)-e

BAD INTERNAL RELOC TABLE
- Rossz belső relokációs tábla

LINK. A programszerkesztő belső hibája.

BAD NUMERIC PARAMETER
(FALSHER NUMERISCH PARAMETER)
- Hibás numerikus paraméter

LINK. A /STACK paraméterben megadott érték nem érvényes numerikus érték.

BAD OR MISSING COMMAND INTERPRETER
(FALSHER ODER FEHLENDER BEFEHLSINTERPRETER)
- Helytelen vagy hiányos parancsértelmező

DOS. Az a lemez, amelyről a DOS indult, nem tartalmazza a COMMAND.COM-ot, vagy a rendszer betöltésénél hiba történt. Ha a problémát a rendszer visszatöltésével nem lehet megoldani, akkor a COMMAND.COM-ot a háttér lemezről ismételten át kell másolni.

Ez az üzenet jelenik meg akkor is, ha a tartalomjegyzékből töröltük a COMMAND.COM bejegyzést. Ezt az üzenetet kapjuk akkor is, ha a COMSPEC = xx paraméter tartalomjegy-

zékhez vezető olyan környezetet ad meg, amely a COMMAND.COM adatállományt nem tartalmazza, és a DOS megkísérli a parancsprocesszor ismételt behívását.

BAD OR MISSING < file-név >
(FALSHER ODER FEHLENDER (DATEINAME))
- Helytelen vagy hiányzó file-név

DOS. Ez az üzenet csak a rendszer indításánál jelenik meg. Azt jelenti, hogy egy periféria-kezelő program, amelyet a COMFIG.SYS file-ban specifikáltunk, nem található. Előfordulhat még, hogy a megadott megszakítási cím (interrupt adress) a meglévő konfiguráció címtartományán kívül esik. Ezt a perifériakezelő programot a DOS nem installálja.

BAD OR MISSING KEYBOARD DEFINITION FILE
- Rossz vagy hiányzó billentyűzet definíciós tábla

KEYB. A program nem találta a KEYBOARD.SYS file-t, vagy az rossz adatokat tartalmazott.

BAD PARTITON TABLE
- Hibás partíciós tábla

FORMAT. A lemez partíciós táblája hibás, vagy nem tartalmaz DOS partíciót.

BAD UNIT
- Hibás meghajtó

Bármelyik parancs küldheti. Valamelyik lemezes perifériakezelő egy kritikus hibát talált valamelyik egységen.

BAD TRACKS FOUND AT START OF PARTITION
PARTITION SIZE ADJUSTED
- Hibás sávok miatt a partíció mérete módosult

FDISK. A fix lemez DOS által igényelt részén hibás blokkok vannak. A rendszer ezeket átlépte, ezzel a partíció mérete csökkent.

BATCH FILE MISSING
(STAPELVERARBEITUNGSDATEI FEHLT)
- A kötegelt (batch) file hiányzik

DOS. A DOS a parancs file-t nem találta az illető .BAT file feldolgozása közben. Ez valószínűleg törlés vagy átnevezés miatt történt. A feldolgozás befejeződik és a DOS rendszerazonosító (prompt) jelenik meg.

BF

DEBUG. Hibás azonosítójel. A kódot hibásan adtuk meg. A regiszter parancsot (RF) a helyes értékkel ismételten ki kell adni.

BP

DEBUG. Teszt leállások (break-points). Több, mint tíz teszt leállást adtunk meg a GO parancs számára. A GO parancsnak legfeljebb tíz teszt leállást adjuk meg.

BR

DEBUG. Hibás regiszter azonosítót adtunk meg. Az R parancsot helyes azonosítóval ismételten ki kell adni.

BREAK IS ON/OFF
- CTRL-BREAK használata BE/KI kapcsolva

BREAK. Az üzenet jelzi, hogy a <CTRL> + <Break> leütésének van-e azonnali hatása vagy sem. Nem hiba.

BUFFER SIZE ADJOUSTED
- Puffer mérete módosítva.

VDISK. A parancs módosította a CONFIG.SYS-ben megadott bufferméretet.

BUFFER SIZE	Puffer méret
SECTOR SIZE	Szektor méret
DIRECTORY ENTRIES	Tartalomjegyzék nagysága
TRANSFER SIZE	Egyszerre átvihető sávok száma

C...

CANNOT CHDIR TO ROOT
- A főtartalomjegyzék sérült

CHKDSK. A CHKDSK parancs végrehajtása félbeszakadt, mert a DOS nem tudott egy CD\ parancsot végrehajtani. A lemez sérült.

CANNOT CHKDSK A NETWORK DEVICE
CANNOT CHKDSK A NETWORK DRIVE
- A parancs hálózaton implementált lemezegységre nem hajtható végre.

CHKDSK.

CANNOT CHKDSK A SUBSTED OR ASSIGNED DRIVE
- SUBST és ASSIGN után a CHKDSK nem használható

CHKDSK. A SUBST parancs elérhetetlenné teszi a CHKDSK számára szükséges információt, ezért az nem hajtható végre.

CANNOT CREATE EXTENDED DOS PARTITION WITHOUT PRIMARY DOS PARTITION ON DISK 1
- Az első fix lemezen nem hozható létre kibővített DOS partíció az elsődleges DOS partíció nélkül.

FDISK. Az 1-es fix lemezen kibővített DOS partíciót akartunk létrehzni, de még nem hoztunk létre elsődlegesDOS partíciót. Először ez utóbbit kreáljuk meg, s csak utána a kibővítettet.

CANNOT CREATE A ZERO CILINDER PARTITION
- 0 cilinderből álló partíció nem hozható létre

FDISK. 0 cilinderből álló partíciót próbáltunk meg létrehozni (valószínűleg tévedésből). Adjuk meg a helyes partícióméretet, és ismételjük meg a funkciót.

CANNOT DELETED EXTENDED DOS PARTITION WHILE LOGICAL DRIVES EXIST
- Logikai egységet tartalmazó kibővített DOS partíciót nem törölhetünk.

FDISK. A kibővített DOS partíció törlése előtt az abban levő valamennyi logikai egységet törölni kell. Ha ezt elfelejtettük, akkor kapjuk ezt az üzenetet.

CANNOT DISKCOMP TO OR FROM A NETWORK DRIVE
- DISKCOMP hálózaton nem használható

DISKCOMP. A parancs nem használható hálózaton implementált lemezegységre még akkor sem, ha az fizikailag ugyanazon a számítógépen van. Ehelyett a COMP *.* parancsot célszerű használni.

CANNOT DISKCOPY TO OR FROM A NETWORK DRIVE
- DISKCOPY hálózaton nem használható

DISKCOPY. A DISKCOPY parancs nem használható hálózaton implementált lemezegységre még akkor sem, ha az fizikailag ugyanazon a számítógépen van. Ehelyett a COPY *.* parancsot lehet használni.

CANNOT DO BINARY READS FROM A DEVICE
(BINARES LESEN VON EINER EINHEIT UNMÖGLICH)
- A bináris olvasás nem hajtható végre az egységről

COPY. A /B paramétert egységnévvel használtuk (Pl. CON:) a másoláshoz. A másolás bináris módban nem hajtható végre, mert a COPY-nak (EOF) jelet kell kapnia az egységről. Ekkor a /B paramétert el kell hagyni, vagy az egységnév után a /A paramétert kell megadni.

CANNOT EDIT .BAK FILE - RENAME FILE
(.BAK-DATEI KANN NICHT EDITIERT WERDEN - DATEI UMBENNEN)
- A .BAK kiterjesztésű file-ok védett állományok, ezért ezek nem szerkeszthetők. Ezeket csak átnevezés után lehet szerkeszteni.

CANNOT EXEC BASICA.COM
- A BASICA.COM file nem tölthető be

BASIC. A ROM BASIC nem tudja elindítani a BASICA.COM file-t. Feltehető, hogy a file nincs az elérhető szótárak egyikében sem.

CANNOT EXECUTE FORMAT
- FORMAT nem hajtható végre

SELECT. A FORMAT végrehajtása közben hiba történt. Elképzelhető, hogy a parancsot nem találja a rendszer.

CANNOT EXECUTE XCOPY
- XCOPY nem hajtható végre

SELECT. Az XCOM végrehajtása során hiba történt. Elképzelhető, hogy a parancsot nem találja a rendszer.

CANNOT FDISK A NETWORK DRIVE
- FDISK hálózaton nem használható

FDISK. A hálózaton implementált fix lemezegységet nem lehet átparticionálni.

CANNOT FIND FILE <OBJEKT FILE> CHANGE DISKETTE AND PRESS <ENTER>
(DATEI NICHT GEFUNDEN - OBJEKTDATEI DISKETTE AUSTAUSCHEN <EINGABE BETTIGEN>)
- A tárgy modul nem található. Cserélje ki a lemezt

LINK. A LINK program a megadott tárgy (objekt) modult nem találta. A megadott modult tartalmazó floppy-t kell betenni, majd az <ENTER> billentyű lenyomásával a LINK folytatja a programszerkesztést.

CANNOT FIND LIBRARY FILE ENTER NEW DRIVE LETTER
(BIBLIOTHEK NICHT GEFUNDEN - BIBLIOTHEKSDATEI NEUE LAUFWERK-BUCHSTABEN EINGEBEN)
- A könyvtári file nem található. Adja meg az új meghajtó azonosítót.

LINK. A megadott könyvtár (.LIB) nem található a lemezen. Azt az új meghajtó azonosítót kell beadni, amelyen a könyvtár található.

CANNOT FIND SYSTEM FILES
- A rendszer file-ok nem találhatók

FORMAT vagy SYS. Az IBMBIO.COM és/vagy IBMDOS.COM file-okat nem találja a program.

CANNOT FORMAT A NETWORK DRIVE
- FORMAT hálózaton nem használható

FORMAT. Hálózaton implementált lemezegységet nem lehet megformázni.

CANNOT FORMAT A SUBSTED OR ASSIGNED DRIVE
- SUBST és ASSIGN után a FORMAT nem használható

FORMAT. A FORMAT parancs kiadása előtt nem használhatjuk a SUBST és ASSIGN parancsokat. Az ASSIGN segítségével állítsuk vissza az eredeti állapotot és utána formázzuk meg a lemezt.

CANNOT FORMAT NONREMOVABLE DRIVE X
- Az x fix egység nem formázható

BACKUP. A parancs használatakor az /F opciót is megadtuk, holott az fix egységre nem használható. Ha mégis szükség van a fix lemez megformázására, azt a BACKUP parancs előtt, a FORMAT-tal végezzük el.

CANNOT JOIN TO A NETWORK DRIVE
- JOIN hálózaton nem használható

JOIN. A JOIN segítségével nem kapcsolhatunk össze lokális és hálózaton implementált lemezegységet.

CANNOT LABEL A NETWORK DRIVE
- LABEL hálózaton keresztül nem használható

LABEL. A parancsot nem adhatjuk ki egy hálózaton implementált lemezegységre.

CANNOT LOAD COMMAND, SYSTEM HALTED
(COMMAND KANN NICHT GELADEN WERDEN, SYSTEM ANGEHALTEN)
- A COMMAND .COM nem tölthető, rendszer leállítva

DOS. A parancsprocesszor ismételt töltése során a DOS megállapította, hogy a lemezterület megsérült. Ezt az üzenetet kapjuk akkor is, ha a parancsprocesszor a COMSPEC=xx környezet-paraméterrel megadott úton nem található. Ebben az esetben a rendszerlemezről ismét töltsük be a DOS-t.

CANNOT NEST RESPONSE FILE
(ANTWORTDATEI KANN NICHT VERSCHACHTELT WERDEN)
- Az automatikus válasz file-t nem tudta besorolni

LINK. A <@FILE-SPEC>-t, az automatikus válasz-file-t használtuk.

CANNOT OPEN LIST FILE
(LISTDATEI KANN NICHT ERÖFFNET WERDEN)
- A lista file-t nem tudta megnyitni

LINK. A tartalomjegyzék vagy a lemez megtelt.

CANNOT OPEN OVERLAY
(ÜBERLAGERUNG KANN NICHT ERÖFFNET WERDEN)
- Az OVERLAY nem végrehajtható.

LINK. A tartalomjegyzék vagy a lemez megtelt.

CANNOT OPEN RESPONSE FILE: < file-név >
(ANTWORTDATEI KANN NICHT ERÖFFNET WERDEN)
- A válasz-file-t nem tudja megnyitni.

LINK. Az automatikus válasz-file-t nem találta

CANNOT OPEN RUN FILE

LINK. A tartalomjegyzék vagy a lemez megtelt.

CANNOT OPEN TEMPORARY FILE
(TEMPORAERE DATEI KANN NICHT ERÖFFNET WERDEN)
- Az ideiglenes file-t nem tudta megnyitni.

LINK. A tartalomjegyzék vagy a lemez megtelt.

CANNOT PERFORM A CYCLIC COPY
- Ciklus másolás nem hajtható végre.

XCOPY. A paraméterek alapján a forrás és a cél részben azonos lenne. Pl. a cél egy alszótár, míg a forrás egy azt tartalmazó másik altartalomjegyzék.

CANNOT RECOVER ENTRY, PROCESSING CONTINUED
- Nem tud belépni a RECOVER-be, az eljárás folytatódik.

CHKDSK. Nem hibaüzenet.

CANNOT RECOVER TO A NETWORK DRIVE

RECOVER. A RECOVER parancs hálózatban implementált lemezegységre nem használható.

CANNOT REOPEN LIST FILE
- A lista file-t nem lehet újra megnyitni

LINK. Az eredeti lemezt nem helyeztük vissza. Újra kell indítani a programszerkesztőt.

CANNOT START COMMAND, EXITING
(COMMAND KANN NICHT GESTARTET WERDEN, BEENDET)
- COMMAND nem indítható, kilépés.

DOS. A parancsprocesszor (COMMAND.COM) második példányának betöltése során megállapította, hogy a FILES = paraméter a konfigurációs adatállományban túl kicsi

értéket vesz fel, vagy túl kicsi a tárméret ahhoz, hogy a COMMAND.COM másolatát abban elhelyezzük.

CANNOT USE FASTOPEN FOR DRIVE x
- Az x egységre nem használhatjuk a FASTOPEN parancsot.

FASTOPEN. A JOIN, SUBST vagy ASSIGN parancsokban szereplő, ill. hálózaton implementált egységekre nem adhatjuk ki a FASTOPEN parancsot.

CANNOT USE PRINT - USE NET PRINT

PRINT. A hálózatvezérlő gépen nem adható ki a PRINT parancs. Helyette a NET PRINT parancsot kell használni.

CANNOT SUBST TO A NETWORK DRIVE

SUBST. A parancs nem használható hálózaton implementált egységre.

CANNOT SYS TO A NETWORK DRIVE
- SYS hálózaton nem használható.

SYS. A SYS segítségével a rendszer file-ok nem másolhatók hálózatban implementált lemezegységről vagy lemezegységre.

CANNOT XCOPY FROM A RESERVED DEVICE
- XCOPY fentartott egységről nem használható.

XCOPY. A forrás file helyett egy logikai egységnevet adtunk meg, a másolás nem hajtható végre.

CANNOT XCOPY TO A RESERVED DEVICE
- XCOPY fenntartott egységre nem használható.

XCOPY. A cél file helyett egy logikai egységnevet adtunk meg, a másolás nem hajtható végre.

CHDIR... FAILED TRYING ALTERNATE METHOD
- Az Alternatíva módszer sikertelennek bizonyult.

CHKDSK. A DOS-t újból kell indítani.

CODE PAGE NOT PREPARED
- A kódlap nincs betöltve

MODE. A MODE parancsban egy olyan kódlap sorszámára hivatkozhatunk, amelyiket vagynem aktiváltunk, vagy a kódlap formája nem felel meg a választott kijelzési módnak.

CODE PAGE xxx NOT PREPARED FOR ALL DEVICES
- Az xxx sorszámú kódlapot nem definiáltuk valamennyi egységre.

CHCP. A parancs nem tudta a jelzett kódlapot valamennyi egységre aktualizálni. Ennek

többféle oka lehet: az egység nem volt felkészítve erre a kódlapra, átvitelihiba történt, az egység nyomtatott, stb. A hiba elhárítása után a parancs megismételhető.

CODE PAGE xxx NOT PREPARED FOR SYSTEM
- az xxx sorszámú kód-lapot a rendszer nem tudja használni

CHCP. A jelzett sorszámú kód-lapot a parancs nem tudja kiválasztani. Ennek több oka lehet: nem töltöttük be a NLSFUNC programot, a kód-lap az adott ország-specifikus információ mellett nem használható, vagy valamelyik eszközt nem készítettük elő a MODE paranccsal.

CODE PAGE OPERATION NOT SUPPORTED ON THIS DEVICE
- Ezen az egységen a kód-lapok nem használhatók.

MODE. A PREPARE vagy SELECT alparancsot olyan egységre használtuk, amelyik nem képes a kód-lapok kezelésére. Ennek az is lehet az oka, hogy a CONFIG.SYS fileban nem megfelelő DEVICE parancsot helyeztünk el.

CODE PAGE REQUESTED yyy IS NOT VALID FOR GIVEN KEYBOARD CODE
- a megadott kód-lap kódja (yyy) az adott billentyűzet-kóddal összeférhetetlen.

KEYB. A KEYB paranccsal olyan kód-lapot kívántunk aktivizálni, amelyik az adott billentyűzet-kóddal nem használható.

CODE PAGE SPECIFIED HAS NOT BEEN DESIGNATED
- a megadott kód-lap nem használható.

KEYB. A parancsban megadott kód-lapot nem definiáltuk. A MODE PREPARE paranccsal ezt tegyük meg, majd ismételjük meg a KEYB-t! Ez az üzenet csak abban az esetben jelenik meg, ha a kód-lapozást a CON:-ra installáltuk.

CODE PAGE SPECIFIED HAS NOT BEEN PREPARED
- a megadott kód-lapot nem készítettük elő.

KEYB. A CON: eszközre installáltuk a kód-lapozást, de a megadott kód-lap használatára nem készítettük elő. A MODE paranccsal ezt végezzük el, majd ismételjük meg a KEYB-t!

CODE PAGE SPECIFIED IS INCONSISTENT WITH INVOKED CODE PAGE
CODE PAGE SPECIFIED IS INCONSISTENT WITH SELECTED CODE PAGE
- a megadott kód-lap összeegyeztethetetlen a meghívott kód-lappal.

KEYB. Figyelmeztetés: a parancsban megadott kód-lap lett az aktuális, de a CON: és a billentyűzet kód-lapja eltér.

CODE PAGE xxx
- kód-lap = xxx

MODE. A MODE segítségével az aktív kód-lapot kérdeztük le. Információs üzenet.

CODE PAGES CAN NOT BE PREPARED
- a kód-lapok nem készíthetők elő.

MODE. A megadott kód-lapok között ugyanaz a sorszám kétszer szerepel, vagy többet adtunk meg a megengedettnél. Ellenőrizzük a kód-lapok sorszámának a beírását, majd ismételjük meg a MODE parancsot.

COMMON AREA LONGER THAN 65536 BYTES
- A közös rész több, mint 65536 byte

LINK. A felhasználói program COMMON változóinak hossza több, mint 64K, ezért a programszerkeztés nem hajtható végre. Jelenleg csak a Microsoft C fordíto okozhatja ezt a hibát.

COMn: bbb,p,d,s,t INITIALIZED
(COMn: bbb,p,d,s,t INITIALISIERT)
- Az aszinkron kommunikációs adaptert a következő értékekkel inicializáltuk

MODE. Az egyes értékek jelentése:
n: adapter azonosító (COM1 vagy COM2)
bbb: átviteli sebesség (band)
p: paritás
e: páros
o: páratlan
n: nincs
s: stopbitek (1 vagy 2)
t: az egység típusa:
P: nyomtató (időtúllépés esetén újra próbálkozik)
-:egyéb egység. Nincs ismétlés.

COMPARE ERROR AT OFFSET XXXXXXX
(VERGLEISCHFEHLER BEI OFFSET XXXXXXX)
- Összehasonlítási hiba az OFFSET XXXXXXX-nél.

COMP. Az összehasonlított file-ok a megadottt helyen (hexadecimális offset) különböző értékeket tartalmaznak. Az eltérő byte-ok hexadecimális formában kijelzésre kerülnek.

COMMAND FORMAT: DISKCOPY d: d: [/1]
- a parancs alakja DISKCOPY d: d: [/1]

DISKCOPY. Érvénytelen paraméterekkel hívtuk meg, s válaszként kiírja a paraméterek használatának a módját.

COM: ,e,7,1
COMPARE ERROR (3) 04
Track xx, side yy
- Eltérés az yy oldal xx sávján.

DISKCOMP. A jelzett helyen egy vagy több blokkban eltérő adatokat talált a parancs a két lemezen.

COMh: bbbb,p,d,s,t, INITIALIZED
- a COMh csatorna a bbbb,p,d,s,t paraméterekkel üzemkész.

MODE.	Az egyes paraméterek jelentése:
h	adapter (COM1-COMh);
bbbb	átvitel sebessége baud-ban;
p	paritás;
e	páros;
o	páratlan;
n	semmilyen;
d	adatbitek;
s	adatbitek (1,2)
t	a soros eszköz típusa
p	soros nyomtató
-	nincs soros eszköz

COMPARE MORE DISKETTES (Y/N)?
(WEITERE DISKETTEN VERGLEICHEN (J/N)?)
- Kell még lemezeket összehasonlítani (I/N)?

DISKCOMP. Ha további összehasonlítást akarunk, akkor Y-t, ill. J-t kell megadni. Ebben az esetben a DISKCOMP kéri az összehasonlítandó lemezek behelyezését. Ha nem akarjuk a lemezeket összehasonlítani, N-t kell megadni.

COMPARE MORE FILES (Y/N)?
- Kell még file-okat összehasonlítani (I/N)?

COMP. A parancs befejezte a file-ok összehasonlítását és megkérdezi, hogy akarunk-e még továbbiakat összehasonlítani. N-re visszatérhetünk a DOS-ba, az Y újraindítja a COMP parancsot.

COMPARE PROCESS ENDED
- az összehasonlítás befejeződött.

DISKCOMP. Információs üzenet: a DISKCOMP parancs végrehajtása befejeződött.

COMPARING X SECTORS PER TRACK, n SIED(S)
(VERGLICHTEN WERDEN X SEKTOREN PRO SPUR, n SEITE(n))
- Összehasonlításra kerül sávonként x szektor, n oldal

DISKCOMP. Az n értéke 1 vagy 2. Az összehasonlítandó oldalak számát adja. Az x a sávonkénti szektorok száma az első lemezen (8 vagy 9). Ha a /8 paramétert megadtuk, akkor ez az érték 8.

CONFIGURATION TOO LARGE FOR MEMORY
- A konfiguráció túl nagy a memóriához képest.

DOS. A CONFIG.SYS parancsban megadott FILES és BUFFERS allokálása után nem maradt elég memória a DOS betöltésére. Előfordulhat, hogy a COMMAND-ban szerepel a /E paraméter , és az abban megadott méret kevés a COMMAND.COM file betöltésére.

CONTAINS INVALID CLUSTER, FILE TRUNCATED
(DATEI HAT UNGÜLTINGEN BEREICH, DATEI ABGESCHNITTEN)
- A file érvénytelen területet tartalmaz, a file-t megcsonkította.

CHKDSK. A file érvénytelen adatblokkot tartalmaz. A hibás blokkot levágja /F paraméter esetén.

CONTAINS XXX NON-CONTIGUOUS BLOCKS
(DATEI ENHÄLT XXX NICHT ZUSAMMENHÄNGENDE BLÖCKE)
- A file nem összefüggő blokkot tartalmaz.

CHKDSK. A megadott file fizikailag nem összefüggő területen helyezkedik el, hanem xxx darabban. Ez informatív üzenet. A felszeletelt file feldolgozása hosszabb időt vesz igénybe, ezért érdemes esetleg egy másik lemezre átmásolni (COPY). Ekkor a file feltehetően kevesebb darabban kerül a lemezre, és így gyorsabb lesz az elérése.

CONVERT DIRECTORY TO FILE (Y/N)?
(INHALTSVERZEICHNIS IN DATEI UMWANDELN (J/N)?)
- A tartalomjegyzék átalakítása a file-hoz (I/N)?

CHKDSK. A megadott tartalomjegyzék már annyi hibás bejegyzést tartalmaz, hogy az már nem használható directory-ként. Ha Y ill. J választ adunk, akkor a rendszer a file-ok listáját tartalomjegyzékké alakítja, és DEBUG programmal átvizsgálhatóvá válik. Ha N-t adunk meg, akkor a bejegyzések nem változnak a tartalomjegyzékben.

CONVERT LOST CHAINS TO FILES (Y/N)?
(VERLORENE BLÖCKE IN DATEINEN UMVANDELN (J/N)?)
- Az elveszett láncok file-okká alakítása (I/N)?

CHKDSK. Ha a hibás blokkokban található file-okat ismét elő akarjuk állítani, akkor Y-t ill. J-t kell megadni. Ha N-t adunk meg, akkor a CHKDSK törli az adathibás blokkokat.

COPY ANOTHER (Y/N)?
(WEITERE KOPIE ERSTELLEN (J/N)?)
- Szükséges még másik COPY (I/N)?

DISKCOPY. Ha újabb teljes lemezt akarunk másolni, akkor Y-t, ill. J-t adunk meg. A DISKCOPY ezután kéri a szükséges lemez behelyezését. Ellenkező esetben N-t kell megadni.

COPY COMPLETE
(KOPIEREN BEENDET)
- A másolás befejeződött.

DISKCOPY. A másolás a forráslemezről a céllemezre sikeresen befejeződött.

COPYING X SECTORS PER TRACK, N SIDE
(KOPIEREN VON X SEKTOREN PRO SPUR, N SEITE)
- Másolás X szektor sávonként, N oldal

DISKCOPY. Az N 1 vagy 2 lehet. Ez az oldalak számát adja meg, amit a forráslemez első sávjáról olvasott le a DISKCOPY. Az X 8 vagy 9 lehet, a forráslemez sávonkénti szektorainak száma.

CURRENT CODE PAGE SETTINGS
- aktív kód-lapok beállítása.

MODE. A MODE GET GLOBAL parancsra válaszként ezzel a szöveggel kezdődő listát kapunt. Információs üzenet.

CURRENT DRIVE IS NO LONGER VALID
- Az aktuális lemezegység hibás.

COMMAND. Miközben a DOS készenléti jelét ($p) akarta kijelezni, a COMMAND érzékelte, hogy az aktuális lemezegység nem tartalmaz érvényes információt, pl. töröltük a hálózaton implementált egységet.

CURRENT KEYBOARD CODE: xx CODE PAGE: yyy
CURRENT CON CODE PAGE: zzz
- aktív billentyűzet-kód: xx kód-lap: yyy
- aktív CON kód-lap: zzz

KEYB. Információs üzenet a használt billentyűzet-kódról és a használt kód-lapokról.

CURRENT KEYBOARD DOES NOT SUPPORT THIS CODE PAGE
- a kiválasztott billentyűzet nem támogatja a megadott kód-lapot.

MODE. Egy MODE PREPARE végrehajtása során kiderült, hogy az adott feltételek mellett ez a kód-lap nem használható.

D...

DATA RECORD TOO LARGE
- A LEDATA rekord túl nagy.

LINK. A LEDATA rekord több, mint 10224 byte hosszú volt. Fordítási hiba, valószínűleg a fordító helytelen működéséből adódik.

DELETE CURRENT VOLUME LABEL (Y/N)?
- Töröljem a régi címkét (I/N)?

LABEL. Ha nem írjuk be az új címkét, csak az <ENTER>-t nyomjuk meg, akkor kapjuk a fenti üzenetet. Ha nem akarjuk a meglevő címket törölni, akkor az N, különben a Y megnyomásával válaszoljunk.

DEVICE ERROR DURING STATUS
DEVICE ERROR DURING PREPARE
DEVICE ERROR DURING SELECT
DEVICE ERROR DURING WRITE OF FONT FILE TO DEVICE
- hiba az egységen.

MODE. A jelzett művelet közben az egységen hiba történt.

DEVICE OR CODE PAGE MISSING FROM FONT FILE
- az egység vagy a kód-lap hiányzik a font-file-ból.

MODE. A PREPARE alparancs végrehajtása során kiderült, hogy a font-file nem megfelelő. A hiba hatására a kód-lapok definiálatlanná válnak, valamennyit újból elő kell készíteni.

DEVICE ddd NOT PREPARED
- a ddd sorszámú kód-lap nincs előkészítve.

MODE. A MODE paranccsal próbáltuk lekérdezni a ddd egység állapotát. Eszerint az egység.egyáltalán nem lett a kód-lapozásra előkészítve.

DF

DEBUG. Dupla jelölés. Ellentétes kódokat használtunk a jelölésre. Regiszter parancs (RF) használatánál egy jelölést csak egyszer lehet felhasználni.

DIR PATH LISTING FOR VOLUME xxxxxxx
- Az xxxxxxx kötet útjai következnek.

TREE. Az üzenet a lemez címkéjét írja ki.

DIRECTORY ENTRIES ADJUSTED
- Tartalomjegyzék nagysága módosítva

VDISK. A VDISK parancs módosította a DEVICE = VDISK.SYS parancsban megadott tartalomjegyzékek maximális számát.

DIRECTORY IS JOINED, TREE PAST THIS POINTNOT PROCESSED
- JOIN-nal összekötött egységre a CHKDSK nem lép át

CHKDSK. A lemezegységek összekapcsolása a tartalomjegyzékek fa-struktúráját megnöveli. A CHKDSK egy ilyen ponthoz ért, de nem kezdte el a fizikailag másik lemez ellenőrzését. Nem hibaüzenet.

DIRECTORY IS TOTALLY EMPTY, NO . OR .., TREE PAST THIS POINT NOT PROCESSED
- Üres altartalomjegyzék . és .. nélkül, a CHKDSK nem lép bele.

CHKDSK. A parancs olyan altartalomjegyzéket talált, amelyikben sem a ., sem a .. be-

jegyzés nem szerepel. Valószínűleg az altartalomjegyzék aktivizálása közben rendszerhiba történt. A RECOVER paranccsal meg lehet kísérelni a tönkrement lemezről a file-ok visszaállítását.

DISC BOOT FAILURE
(BOOT FEHLER)
- BOOT hiba

DOS. A DOS memóriába töltése során hiba fordult elő. Ha a rendszer a további próbálkozások során sem indul, akkor egy tartalék rendszerlemezt kell használni!

DISC ERROR READING DRIVE X
- Az x egységen olvasási hiba

A program megpróbált egy fizikai olvasást egy hálózaton implementált lemezegységről. Erre nincs lehetőség.

DISC ERROR WRITING DRIVE X
- Az x egységen íráshiba.

A program megpróbált egy fizikai írást egy hálózaton implementált lemezegységre. Erre nincs lehetőség.

DISC ERROR WRITING FAT X
(PLATTENFEHLER BEI SCHREIBEN DER DATEIZUORDNUNGSTABELLE X)
- Lemez hiba az x FAT írásánál.

CHKDSK. A file-leíró táblázat (FAT) aktualizálása során a CHKDSK lemezhibát talált. Az x 1 vagy 2 lehet aszerint, hogy melyik példányát nem tudta a FAT-ba beírni. Ha ez az üzenet kétszer megjelenik, akkor a lemezt használhatatlannak kell tekinteni. (Lehetséges, hogy a FAT mindkét példánya sérült.)

DISC FULL - WRITE NOT COMPLETED
DISK FULL. EDITS LOST
(DISKETTE VOLL - SCHREIBEN NICHT BEENDET)
- A lemez megtelt - a kiírás nem fejeződött be.

EDLIN. A szerkesztés rendhagyóan fejeződött be, mivel a lemezen nincs elég terület a teljes file kimentésére. A file lemezre mentett része megmarad. de a memóriában levő rész elvész.

DISC NOT COMPATIBLE
(DISKETTE NICHT VERTRÄGLICH)
- A lemez nem kompatibilis

FORMAT. A felhasználó által megadott meghajtón levő lemez a FORMAT paranccsal nem formázható. Ennek oka, hogy a használt lemezegység nem rendelkezik IBM interface-szel.

DISC UNSUITABLE FOR SYSTEM DISC
(DISKETTE NICHT ALS SYSTEMDISKETTE VERWENDBAR)
- A lemez rendszerlemezként nem használható.

FORMAT. A lemez elején hibás sáv található. A lemez DOS rendszer tárolására nem alkalmas, csak adatokéra.

DISKETTE/DRIVE not COMPATIBLE
- A lemez vagy a lemezegység nem kompatibilis.

DISKCOPY vagy DISKCOMP. A cél és forráslemez eltérő formája nem teszi lehetővé a másolás (összehasonlítás) végrehajtását. Pl. az egyik 360 Kbyte-os, míg a másik 1.2 Mbyte-os.

DISKETTES COMPARE OK
(DISKETTENVERGLEICH ERFOLGREICH BEENDET)
- Lemez összehasonlítás OK.

DISKCOMP. Az összehasonlított lemeztartalmak megegyeznek.

DISKETTE IS NOT A BACKUP DISKETTE
(DISKETTE IST KEINE SICHERUNGSDISKETTE)
- A disc nem tartalék disc

BACKUP és RESTORE. A lemez nem BACKUP programmal lett előállítva. A tartalék (backup) lemezen az első file-azonosító a BACKUPID.@@@. ismételjük meg a végrehajtást a megfelelő lemezzel.

DIVIDE OVERFLOW
(TEILERÜBERLAUF)
- Túlcsordulás az osztás műveletben

DOS. A programban 0-val való osztás fordult elő, vagy olyan belső hibát tartalmaz, amely a program megszakításához vezetett. A rendszer programmegszakítást szimulál.

DOES <út> SPECIFY A FILENAME OR DIRECTORY NAME ON THE TARGET (F = FILE, D = DIRECTORY)?
- Az <út>-ról el kell dönteni, hogy file-név, vagy altartalomjegyzék neve (F = file-név, D = altartalomjegyzék).

XCOM. Az üzenet az <út>-ról nem tudja eldönteni, hogy file-nevet, vagy altartalomjegyzéket definiál. Ha D-vel válaszolunk, akkor az XCOPY a cél lemezen létrehozza az adott nevű altartalomjegyzéket.

DO YOU SEE THE LEFTMOST 0 (Y/N)?
(IST DIE GANZ LINKS STEHENDE 0 ZU SEHEN (J/N)?)
- Látja a legbaloldalibb 0-t (I/N)?

MODE. A MODE parancsot a /R /T paraméterrel adtuk meg. A rendszerüzenet addig ismétlődik, amíg Y-t, ill. J-t nem adunk.

DO YOU SEE THE RIGHMOST 9 (Y/N)?
(IST DIE GANZ RECHTS STEHENDE 9 ZU SEHEN (J/N)?)
- Látja a legjobboldalibb 9-t (I/N)?

MODE. A MODE parancsot a /L /T parraméterrel adtuk meg. A rendszerüzenet addig ismétlődik, amíg Y-t, ill. J-t nem adunk.

DO YOU WISH TO USE THE ENTIRE FIXED DISC FOR DOS (Y/N)...?[d]
DO YOU WISH TO USE THE MAXIMUM SIZE FOR A DOS PARTITION AND MAKE THE DOS PARTITION ACTIVE (Y/N)...[]
(SOLL DIE GESAMTE FESTPLATTE FÜR DOS BENUTZT WERDEN (J/N)...?[d]
- A teljes fix lemezt a DOS számára foglalja le (I/N)...?[d]

FDISK. A DOS partíció előállítása lehetőséget választottuk és a fix lemez nem volt még felosztva, akkor jelenik meg ez a kérdés. Ha N-t adunk meg, akkor meg kell adni az előállítandó DOS partíció határait is.

COMMAND FORMAT: DISKCOPY d: d: [/1]
- A parancs alakja: DISKCOPY d: d: [/1]

DISKCOPY. A program érvénytelen paramétert talált, s válaszul kijelezte a parancs helyes alakját.

DOS PARTITION ALREADY EXISTS
- A DOS partíció már létezik

FDISK. A fix lemezegység partíciós tablója már tartalmaz egy DOS partíciót. Mégegyet nem hozhatunk létre.

DOS PARTITION CREATED
- A DOS partíciót létrehozta

FDISK. Információs üzenet: a parancs a fix lemezen létrehozta a DOS partíciót.

DOS PARTITION DELETED
- A DOS partíciót törölte

FDISK. Információs üzenet: a fix lemezen már nincs DOS partíció.

DRIVE d: ALREADY DELETED
- a d: egységet már töröltük.

FDISK. Információs üzenet: a képernyőn a parancs jelzi, hogy melyik logikai egységet töröltük már.

DRIVE LETTER MUST BE SPECIFIED
- Az egység betűjelét meg kell adni

FORMAT. A parancs a 3.20-as verziótól nem használható a lemezegység azonosítójának megadása nélkül.

DRIVE LETTERS HAVE BEEN CHANGED OR DELETED
- a logikai egységekhez rendelt betűjelek megváltoztak, vagy törlődtek.

FDISK. A kibővített DOS partícióból töröltünk egy, vagy több logikai egységet. Ennek hatására a meglevők módosultak.

DRIVE TYPES OR DISKETTE TYPES NOT COMPATIBLE
- Az egységek vagy lemezek típusa eltérő

DISKCOPY vagy DISKCOMP. A cél és forrás lemez típusa összeférhetetlen.

DUP RECORD TOO COMPLEX
- A DUP rekord túl bonyolult

LINK. A DUP rekord túl bonyolult ahhoz, hogy a LINK fel tudja dolgozni. Egyetlen DUP 1024 byte-ot foglal le a kiterjesztés előtt. A DUP-ot tartalmazó sort egyszerűsíteni kell.

DUPLICATE FILENAME OR FILE NOT FOUND
(DOPPELTER DATEINAME ODER DATEI NICHT GEFUNDEN)
- Duplikált vagy nem található file

RENAME. Egy olyan azonosítót akartunk adni a file-nak, amely már létezik, vagy az átnevezendő adatállomány nem található a lemezen.

E...

ENTER CURRENT VOLUME LABEL FOR DRIVE d: (PRESS ENTER FOR NONE):
- Kérem a d: egység kötetazonosítóját: (< ENTER > ha nincs)

FORMAT. A 3.20-as verziótól a fix lemezegység csak akkor formázható meg, ha beírjuk a címkéjét. Ha ez nem jó, a parancs nem hajtódik végre.

ENTER THE NUMBER OF THE PARTITION YOU WANT TO MAKE ACTIVE...:[]
(GEBEN SIE DIE ZU AKTIVIERENDE PARTITION EIN...[])
- Adja meg az aktivizálandó partíciót...:[]

FDISK. Az aktív partíció változtatásának lehetőségét kértük, ezért most meg kell adni az aktivizálandó partíció számát (azonosítóját). Az azonosító a rendszerüzenetben visszajelzésre kerül, majd az < ENTER > billentyűt kell lenyomni.

ENTER PARTITON SIZE...:[DDDD]
(AUSGABE DER PARTITIONSGRÖBE...:[DDDD])
- Adja meg a partíció méretét...:[DDDD]

FDISK. A DOS partíció beállítása során jelenik meg ez az üzenet. A DOS partíció meretét kell megadni sávokban. A szögletes zárójelek között megjelent szám a standard értéket jelöli. Ha csak az üzenet végét jelző billentyűt nyomjuk le, akkor ez lesz a partíció mérete.

ENTER PRIMARY FILE NAME
(EINGABE DES PRIMÄREN DATEINAMENS)
- Adja meg a file-azonosítót

COMP. Az első file azonosítójának kérése.

ENTER 2ND FILE NAME OR DRIVE ID
- Adja meg a második file nevét vagy a lemezegységet

COMP. A második file azonosítójának kérése.

ENTER STARTING CYLINDER NUMBER...:[dddd]
(STARTZYLINDER ANGEBEN...:[dddd])
- Adja meg a kezdő cylinder számát...:[dddd]

FDISK. A DOS partíció előállítása lehetőség felkéri a felhasználót, hogy adja meg a partíció kezdetének cylinder számát. A szögletes zárójelben a legelső lehetséges értéket adja meg a program, egyébként a cylinder kezdő címét kell beadni.

ENTRY ERROR (EINGABEFEHLER)
- Parancs megadási hiba

EDLIN. Az utolsó parancsban található szintaktikai hibát javítani kell.

ENTRY HAS A BAD ATTRIBUTE OR SIZE OR LINK
(EINTRAG HAT FALSCHES ATTRIBUT (GRÖBE ODER VERBINDUNG))
- A belépési attributum rossz (méret vagy kapcsolat)

CHKDSK. Az üzenet egy vagy két ponttal kezdődhet. Azt adja meg, hogy az altartalombejegyzés rossz. A /F esetén a CHKDSK megpróbálja javítani a hibát.

EOF MARK NOT FOUND
(EOF MARKIERUNG NICHT GEFUNDEN)
- Az EOF jel nem található

COMP. A file-(ok) utolsó blokkjában nem sikerült az érvényes adatok végét megtalálni. Valószínű, hogy nem szöveges file-okat hasonlítottunk össze. A részletes információ a COMP parancsnál megtalálható.

ERROR DURING READ OF FONT FILE
- hiba a font-file olvasása közben.

MODE. A PREPARE alparancs végrehajtása során a font-file olvasása közben hiba történt.

ERROR FOUND, F PARAMETER NOT SPECIFIED. CORRECTIONS WILL NOT BE WRITTEN TO DISK
(FEHLER, PARAMETER F NICHT ANGEGEBEN. KORREKTUREN WERDEN NICHT AUF PLATTE GESCHRIEBEN)
- Hiba fordult elő, az F paraméter nem volt megadva, a javítást nem fogja a lemezre írni

CHKDSK. A /F paramétert nem használtuk. A CHKDSK az analízist úgy hajtja végre, hogy a felfedezett hibákat átlépi, így a felhasználó az analízis eredményeit láthatja. A javítások nem íródnak a lemezre.

ERROR IN COUNTRY COMMAND
- hiba a COUNTRY parancsban.

DOS. A COUNTRY parancsban megadott forma nem megfelelő, vagy a megadott file inkorrekt információt tartalmaz.

ERROR IN EXE FILE
(FEHLER IN EXE-DATEI)
- Hiba az EXE file-ban

DOS. A szerkesztett program megváltozott, ezért néhány eltolási-érték (displacement) hibás.

ERROR IN EXE/HEX FILE (FEHLER IN EXE/HEX DATEI)
- Hiba az EXE/HEX file-ban

DEBUG. A file érvénytelen jeleket tartalmaz.

ERROR LOADING OPERATING SYSTEM
(FEHLER BEI LADEN DES BETRIEBSSYSTEMS)
- Hiba az operációs rendszer betöltésének folyamán

START. A fix lemezről való töltés során hiba keletkezett. Ha a hiba ismételten előfordul, akkor a rendszert a tartalék DOS lemezről kell betölteni, és a SYS paranccsal az új másolatát kell létrehozni a fix lemezen.

ERROR OPENING LOG FILE
- hiba a LOG-file megnyitása közben.

BACKUP. A /L opciót használtuk, de a LOG-file létrehozása, vagy megnyitása közben hiba történt.

ERROR READING FIXED DISK (FEHLER BEIM LESEN DER FESTPLATTE)
- Hiba a fix lemez olvasása során

FDISK. A program az aktuális fix lemez kezdeti betöltő rekordját öt kísérlettel sem tudta beolvasni. Meg kell kísérelni az FDISK ismetelt végrehajtását. Ha ez nem sikerül, akkor kapcsolatot kell keresni a PC szállítójával.

ERROR WRITING FIXED DISK
(FEHLER BEIM SCHREIBEN AUF FESTPLATTE)
- Hiba a fix lemezre íráskor

FDISK. A program az aktuális fix lemez kezdeti betöltő rekordját öt kísérlettel sem tudta felírni. Meg kell kísérelni az FDISK ismételt végrehajtását. Ha ez nem sikerül, akkor kapcsolatot kell keresni a PC szállítójával.

ERROR WRITING TO DEVICE
(FEHLER BEI SCHREIBEN AUF EINHEIT)
- Hiba az egységre íráskor

PARANCSOK. A DOS a megadott számú byte-ot nem tudta az egységre írni. A felhasználó több adatot akart átküldeni a lemezre, mint amennyit az várt.

ERRORS ON LIST DEVICE INDICATE THAT IT MAY BE OFF-LINE. PLEASE CHECK
(DIE AUSGABEEINHEIT IST EVENTUELL AUSGESCHALTET ODER OFFLINE. BITTE ÜBERPRÜFEN)
- A listázó egység kikapcsolt (OFF line) állapotban van. Kérem ellenőrizni

PRINT. A háttér nyomtató nincs bekapcsolva. Akkor kapjuk ezt az üzenetet, ha ki van kapcsolva az egység, és új PRINT parancsot adtunk ki.

ERROR READING PARTITION TABLE
- Partíciós tábla olvasási hibás

FORMAT. A fix lemez partíciós táblája nem olvasható, valószínűleg fizikailag hibás. Próbáljuk meg az FDISK programot újra futtatni.

ERROR WRITING PARTITION TABLE
- A partíciós tábla írási hibás

FORMAT. A fix lemez partíciós táblája nem írható, valószínűleg fizikailag hibás. Próbáljuk meg az FDISK programot újra futtatni.

EXE AND HEX FILES CANNOT BE WRITTEN
(EXE UND HEX-DATEINEN KÖNNEN NICHT GESCHRIEBEN WERDEN)
- Az EXE és a HEX file-okra nem lehet írni

DEBUG. A DEBUG nem támogatja .EXE és .HEX kiterjesztésű file-ok kiírását. Más névvel írjuk ki a file-t.

EXTENDED DOS PARTITION ALREADY EXISTS
- a kiterjesztett DOS partíció már létezik.

FDISK. A fix-lemezen egynél több kiterjesztett DOS partíciót nem tudunk létrehozni.

EXTENDED DOS PARTITION CREATED
- a kiterjesztett DOS partíció létrehozása megtörtént.

FDISK. A fix-lemezen létrehoztunk egy kiterjesztett DOS partíciót. Ennek használatához logigai egységeket kell definiálnunk.

EXTENDED DOS PARTITION DELETED
- a kiterjesztett DOS partíció törölve.

FDISK. Információs üzenet.

EXTENDER CARD SWITCHES DO NOT MATCH THE SYSTEM MEMORY SIZE
- A memória bővítő kártya kapcsolói rosszul tükrözik a memória méretét

VDISK. A memóriabővítő kártya mikrokapcsolóit nem megfelelően állítottuk be, és nem jól tükrözi a memória nagyságát.

EXEC FAILURE (FEHLER BEI EXEC)
- Végrehajtási hiba

PARANCSOK. A parancs végrehajtása során hiba fordul elő, vagy a FILES = xx értéknek a konfigurációs file-ban (CONFIG.SYS) túl kicsi értéket adtunk meg. Ezt az értéket meg kell növelni, és a DOS-t ismételten be kell tölteni.

F...

FAILURE TO ACCESS CODE PAGE FONT FILE
- a kód-lap font-file-jának sikertelen elérése.

MODE. A PREPARE alparancs nem tudta a font-file-t használni. Vagy rosszul írtuk be a file nevét, vagy megsérült a file.

FAILURE TO ACCESS COUNTRY.SYS
- a COUNTRY.SYS file sikertelen elérési kísérlete.

SELECT. A COUNTRY.SYS file megnyitása sikertelen, ezért a SELECT parancs abortál.

FAILURE TO ACCESS DEVICE: xxx
- az xxx egység elérése sikertelen.

MODE. Valamelyik kód-laphoz tartozó művelet végrehajtása során az xxx egységen hiba történt. Ellenőrizzük az egység nevét, vagy a CONFIG.SYS file-ban a megfelelő DEVICE parancsot. Ezután újra adjuk ki a MODE parancsot.

FAILURE TO ACCESS KEYBOARD.SYS
- a KEYBOARD.SYS file elérése sikertelen.

SELECT. A parancs nem tudta megnyitni a KEYBOARD.SYS file-t, ezért abortált. Ellenőrizzük, hogy a file rendben a kijelölt úton található-e.

FASTOPEN ALREADY INSTALLED
- a FASTOPEN program már a memóriában van.

FASTOPEN. A parancs tranziens részét másodszor is a memóriába akartuk tölteni. Ez lehetetlen, helyette a fenti üzenetet kapjuk.

FASTOPEN INSTALLED
- a FASTOPEN programot betöltöttük.

FASTOPEN. Információs üzenet.

FATAL: INTERNAL STACK FAILURE, SYSTEM HALTED
- Fatális hiba: nincs hely a veremben, a rendszer leállt

A rendszer üzemelése közben nem volt elég hely a veremben. Ennek oka: feltehetően gyorsan ismetlődő hardware megszakítások, amelyeket a rendszer már nem tud kezelni. Tipikusan ilyen hibát okozhat egy rossz billentyűzet. Ha mégis szükség van a megszakításokra (pl. egy analóg mérőkártya esetén), akkor módosítsuk a verem méretet a STACKS konfigurációs paranccsal.

FILE ALLOCATION TABLE BAD, DRIVE X ABORT, RETRY, IGNORE?
(FEHLERHAFTE DATEIZUORDNUNGSTABELLE, LAUFWERK X ABBRUCH, WIEDERHOLEN, IGNOIEREN?)
- Az x meghajtón lévő lemez file-allokációs táblázata (FAT) rossz, A, R, I?

DOS. Az első felhasználói válasz: R ill. W. Ha a hiba ismételten előfordul, a lemez használhatatlan, újra kell formázni.

<FILE> AND <FILE>
(<FILE-SPEC> AND <FILE-SPEC>)
- <file-specifikáció> és <file-specifikáció>

COMP. Ha a COMP parancsot teljes tartalomjegyzékre (altartalomjegyzék) adjuk ki, akkor az összehasonlítás előtt a két file specifikációját (meghajtó, út) is kiírja.

FILE xxx CANCELED BY OPERATOR
(DATEI VON BEDIENER GESTRICHEN)
- A file-t törölte az operátor

PRINT. Ez az üzenet a nyomtatóra kerül, miután a nyomtatás megszakadt. Jelzi, hogy a nyomtatás nem teljes.

FILE CANNOT BE CONVERTED
- A file-t nem lehet konvertálni

EXE2BIN. A konvertálni kívánt file struktúrája nem megfelelő, ezért a .COM file-t a parancs nem tudja előállítani.

FILE CANNOT BE COPIED ONTO ITSELF
(DATEI KANN NICHT IN SICH SELBST KOPIERT WERDEN)
- A file nem másolható önmagára

DOS. Egy file-t másolni akarunk. A másolatot ugyanazzal a névvel ugyanabba a tartalomjegyzékbe akarjuk tölteni az eredeti lemezen. Ebben az esetben a másolat nevét meg kell változtatni, és/vagy más tartalomjegyzékbe vagy lemezre kell másolni.

FILE CREATION ERROR
(FEHLER BEIM ERSTELLEN DER DATEI)
- File előállítási hiba

DOS és PARANCSOK. Nem sikerült a tartalomjegyzékben új file-nevet elhelyezni, vagy a már meglevő file-t felülírni. Ha már létező file-ról van szó, akkor az csak olvasásra érhető el, ezért nem lehetett felülírni. Különben CHKDSK parancsot kell végrehajtani, hogy megállapítsuk, tele van-e a tartalomjegyzék, vagy más körülmény vezetett-e a hibához.

FILE IS CROSS-LINKED : ON CLUSTER XX
(DATEI HAT QUERVERWEIS IM BEREICH XX)
- A file fordított kapcsolást tartalmaz

CHKDSK. Ezt az üzenetet mindig kétszer kapjuk meg hibás láncolás esetén. Ekkor mindkét file új azonosítót kap. Mindkét file-hoz ugyanaz az adatblokk rendelődik. A hiba kiküszöbölése nem automatikus. A felhasználó ezt a következöképpen teheti.

- Mindkét file másolása COPY paranccsal
- Az eredeti file törlése az ERASE paranccsal
- A file-ok tartalmának javítása és szerkesztése

<FILE> IS CURRENTLY BEING PRINTED / <FILE> IS IN QUEUE
(DATEI WIRD GEDRUCKT / DATEI STEHT IN WARTESCHLANGE)
- A file nyomtatóra került / a file várakozási sorban van

PRINT. Ezt az üzenetet akkor kapjuk, ha a PRINT parancsot paraméterrel adtuk meg. (Egy vagy több file-azonosítót a várakozási sorba /queue/ helyeztünk.) Ez az üzenet a felhasználó informálását szolgálja.

FILE IS READ-ONLY
- A file csak olvasható

EDLIN. A megadott file csak olvasható.

FILENAME MUST BE SPECIFIED
- A file-nevet meg kell adni

EDLIN. Az EDLIN parancsból hiányzik a szerkesztendő file neve.

FILE NOT FOUND
(DATEI NICHT GEFUNDEN)
- A file-t nem találja

DOS és PARANCSOK. A file azonosítója a megadott tartalomjegyzékben nem szerepel.

FILE NOT IN PRINT QUEUE
- A file nincs a várakozási sorban

PRINT. A törölni kívánt file nincs a várakozási sorban.

FILE SHARING CONFLICT
- File megosztási hiba

COMP. A parancs nem tudja összehasonlítani a file-okat, mert egyikük osztott és egy másik processz használja.

FILES ARE DIFFERENT SIZES
(DATEINEN HABEN UNTERSCHIEDLICHE GRÖBEN
- A file-ok különböző méretűek

COMP. Az összehasonlítandó file-ok mérete különböző. Az összehasonlítást nem végzi el.

FILES COMPARE OK
(DATEIENVERGLEICH OK)
- File-ok összehasonlítása OK

COMP. Az összehasonlított két file azonos információkat tartalmaz.

FILES WERE BACKED UP XX/XX/XX
- A file-okat xx/xx/xx-kor mentettük

RESTORE. Tájékoztatás a mentés dátumáról.

FIRST CLUSTER NUMBER IS INVALID, ENTRY TRUNCATED
(ERSTE BEREICHSNUMBER UNGÜLTIG, EINGABE ABGESCHNITTEN)
- Az első területszám érvénytelen, a parancs beadása befejeződött

CHKDSK. A megadott file érvénytelen mutatót tartalmaz. A file 0 hosszúságúra lesz levágva.

FIXED BACKUP DEVICE d: IS FULL
- A mentésre használt fix lemez megtelt

BACKUP. A mentés fix lemezegységre történt, ami megtelt. A BACKUP parancs befejezi a mentést.

FIXUP OFFSET EXCEEDS FILELD WITH
(KORREKTUROFFSET ÜBERSCHREITET DIE FELDLÄNGE)
- A megadott offset túllépi a mező szélességét

LINK. Az assembler utasítás egy NEAR attributummal rendelkező címre vonatkozik egy FAR attributummal rendelkező helyett. Az assembler forrásprogramot javítani kell, majd újra kell fordítani.

FIXUPS NEEDED - BASE SEGMENT /HEX/:
(KORREKTUR ERFODERLICH - BASISSEGMENT /HEX/)
- Javítás szükséges. A bázis szegmens /HEX/:

EXE2BIN. A forrás (input) file (.EXE kiterjesztés) olyan információkat tartalmaz, amelyekkel azt adjuk meg, hogy a file-t szegmensenként kell tárolni. Meg kell adni az abszolút szegmenscímet, amelyre a modult betöltjük.

Figyelmeztetés!

Mivel a szegmenst egy meghatározott címre kell tölteni, ezért az ilyet xxxxxxxx.COM fileként kell meghatározni.

FONT FILE CONTENTS INVALID
- a font-file tartalma rossz.

MODE. A font-file feldolgozása során a parancs érzékelte, hogy rossz file-t olvasott be. Lehet, hogy nem font-file-t adtunk meg, vagy a file megsérült.

FOR CANNOT BE NESTED
(FOR KANN NICHT VERSCHACHTELT WERDEN)
- A FOR parancsokat nem szabad egymásba skatulyázni

PARANCS FILE-OK. A parancs file egy sorában több FOR alparancs is található. Egy sorban csak egy FOR megengedett.

FORMAT FAILURE
(FORMATIERFEHLER)
- Formázási hiba

FORMAT. A céllemez előállítása során lemezhibát találtunk. A lemez használhatatlan.

FORMAT NOT SUPPORTED ON DRIVE d:
- A d: egységet nem lehet megformázni

FORMAT. A lemezegység kezelő program (driver) nem, vagy rosszul támogatja az általános IOCTL hívásokat. A VDISK memória lemezei nem formázhatók meg.

FORMATTING WHILE COPYING
(FORMATIEREN WÄHREND KOPIEREN)
- Másolás alatti formázás

DISKCOPY. A céllemez nem formázott. A DISKCOPY a formázatlan területet az átvitel folyamán formázza. Ha ezt az üzenetet nem megfelelő meghajtó típus üzenet követi, akkor ez azt jelöli, hogy a kétoldalú lemezt egyoldalú meghajtón akartuk másolni. A feldogozás befejeződik. A céllemez hibás adatokat tartalmaz.

G...

GRAPHIC CHARACTERS LOADED
(GRAFIKZEICHEN GELADEN)
- A grafikus karaktereket betöltötte

GRATABL. A GRAFTABL.COM file-t betöltötte.

H...

HARDWARE CODE PAGES:
PREPARED CODE PAGES:
- hardware és betöltött kód-lapok.

MODE. A parancs segítségével a kód-lapokra vonatkozó információkat kértük a képernyőre. A fenti fejlécek kiírása után a parancs kilistázza a megfelelő értékeket.

HAS INVALID CLUSTER, FILE TRUNCATED
- Hibás blokk, a file csonkolt lesz

CHKDSK. Az üzenet előtt kiút nevű file érvénytelen mutatót tartalmaz. A /F paraméter használatával meg kell kísérelni a végét csonkolni.

HEAD : h CYLINDER: c
- Fej:h CILINDER: c

FORMAT. Az üzenettel nyomon lehet követni, hogy hol tart a lemez megformázása. Nem hibaüzenet.

I...

ILLEGAL DEVICE NAME (UNGÜLTIGER EINHEITENNAME)
- Helytelen periféria azonosító

MODE. A parancsban megadott paraméterek a következők lehetnek: nyomtató: LPT1, LPT2,LPT3; aszinkron átviteli vonal-paraméter: COM1, COM2; a MODE és a paraméterek között legfeljebb egy üres jel állhat.

INCOMPATIBLE DISKETTE OR DRIVE TYPES
(UNVERTRÄGLICHE DISKETTEN-ODER LAUFWERKART)
- Nem kompatibilis lemez vagy meghajtó típus

DISKCOMP. Az első lemez mindkét oldalát sikerült elolvasni. A második lemeznek viszont csak a második oldalát sikerült. Ebben az esetben vagy a második lemez, vagy a meghajtó csak egyoldalas, vagy az első lemez sávonként 9, a második sávonként 8 szektort tartalmaz.

INCOMPATIBLE DRIVE TYPES (UNVERTRÄGLICHE LAUFWERKARTEN)
- Nem kompatibilis meghajtó típus

DISKCOMP. Az első lemez és meghajtó kétoldalas, a második lemezt tartalmazó viszont csak egyoldalas.

INCOMPATIBLE SYSTEM SIZE (UNVERTRÄGLICHE SYSTEMGRÖBE)
- Nem elfogadható rendszerméret

SYS. A céllemez olyan DOS rendszert tartalmaz, amely kisebb, mint az éppen másolandó lemez. A rendszer átvitele nem hajtható végre. Lehetséges megoldás: a FORMAT/S parancs segítségével egy üres lemezt formázunk, és a file-okat az új lemezre másoljuk.

INCORRECT APPEND VERSION
- rossz APPEND verzió.

APPEND. A program rossz verzióját használtuk. Az IBM hálózati rendszeréhez tartozó APPEND a DOS-szal nem képes együtt dolgozni.

INCORRECT DOS VERSION
(FALSCHE DOS-VERSION)
- Helytelen DOS verzió

PARANCSOK. A felhasználó által kiadott DOS parancs a DOS egy másik verzióját használja. Töltsük be a DOS megfelelő verzióját, vagy használjuk a parancs megfelelő verzióját.

INCORRECT DOS VERSION, USE DOS 2.00 OR LATER
- Rossz DOS verzió, csak 2.00 vagy újabb használható

LINK. A LINK a DOS 2.00-nál korábbi verzióival nem fut. Használjuk a megfelelő verziót!

INCORRECT NUMBERS OF PARAMETERS
- A paraméterek száma rossz

A legtöbb parancs küldheti ezt az üzenetet: a paraméterek száma nem megfelelő.

INCORRECT PARAMETER
- Rossz paraméter

SHARE. A paracsban megadott valamelyik paraméter rossz.

INFINITE RETRY ON PARALLEL PRINTER TIME-OUT
- A nyomtatón végtelen sokszor próbálkozik a rendszer

MODE. Tájékoztató üzenet: a P paramétert adtuk meg, ami lehetővé teszi a végtelen ideig történő próbálkozást.

INFINITE RETRY NOT SUPPORTED ON NETWORK PRINTER
- Hálózati nyomtatón nem lehet végtelen sokszor próbálkozni

MODE. A P paramétert használtuk. Ez nem megengedett, mert a hálózaton implementált nyomtató hibája nem kérdezhető le.

INSERT BACKUP DISKETTE XX IN DRIVE D STRIKE ANY KEY WHEN READY
(SICHERUNGSDISKETTE XX EINLEGEN IN LAUFWERK, WENN BEREIT, EINE TASTE BETÄTIGEN)
- Helyezze az xx háttér lemezt a meghajtóba, ha kész, nyomjon meg egy billentyűt

RESTORE. A következő mentett lemez behelyezése után a RESTORE a feldolgozást akkor folytatja, ha egy tetszőleges billentyűt lenyomunk.

INSERT BACKUP DISKETTE XX IN DRIVE D. WARNING! DISKETTE FILES WILL BE ERASED. STRIKE ANY KEY WHEN READY
(SICHERUNGSDISKETTE XX EINLEGEN IN LAUFWERK. ACHTUNG! DATEN WERDEN ZERSTÖRT. WENN BEREIT, EINE TASTE BETÄTIGEN)
- Az xx lemezt helyezze a meghajtóba. Figyelem! Az adatok törlődnek. Ha kész, nyomjon le egy billentyűt.

BACKUP. A következő mentési lemezt kell elhelyezni a meghajtóban. Erre csak DOS által formázott lemez használható. Egy tetszőleges billentyű lenyomásával folytatható a mentés.

INSERT BACKUP SOURCE DISKETTE IN DRIVE D STRIKE ANY KEY WHEN READY
- Helyezze be a mentendő lemezt az egységbe, utána nyomjon meg egy billentyűt

BACKUP. A rendszer a mentendő lemezt kéri. Behelyezés után egy tetszőleges billentyűt kell megnyomni.

INSERT BACKUP TARGET DISKETTE IN DRIVE D STRIKE ANY KEY WHEN READY
- Helyezze be a cél lemezt az egységbe, utána nyomjon meg egy billentyűt

BACKUP. A rendszer a cél mentési lemezt kéri. Behelyezés után egy tetszőleges billentyűt meg kell nyomni.

INSERT COMMAND.COM DISK IN DRIVE D AND STRIKE ANY KEY, WHEN READY
(COMMAND.COM EINLEGEN IN LAUFWERK D WENN BEREIT, EINE BELIEBIGE TASTE BETÄTIGEN)
- Helyezze a COMMAND.COM-ot tartalmazó lemezt a meghajtóba, ha kész, nyomjon le egy tetszőleges billentyűt

DOS. A DOS megkísérli a parancsprocesszor (COMMAND.COM) ismételt betöltését, de a parancsprocesszor nem található azon a meghajtón, amelyről a DOS indult. A DOS lemezt be kell tenni a meghajtóba, majd egy tetszőleges billentyűt le kell nyomni.

INSERT DISK TO BE RECOVERED INTO DRIVE X: AND PRESS ANY KEY WHEN READY
(WIEDERHERZUSSTELLENDE DISKETTE EINLEGEN IN LAUFWERK D WENN BEREIT, EINE TASTE BETÄTIGEN)
- Tegye a meghajtóba a visszaállítandó lemezt, majd nyomjon meg egy tetszőleges billentyűt

RECOVER. INSERT DISKETTE FOR DRIVE D AND PRESS ANY KEY WHEN READY
- Helyezzen be egy lemezt az D meghajtóba, utána nyomjon meg egy billentyűt

DOS. Logikai lemezek használatakor a rendszer a meghajtóba kér egy lemezt. Ha nem a kívánt lemez van a meghajtóban, helyezzük be a megfelelőt, majd nyomjuk meg akármelyik billentyűt.

INSERT DISK WITH BATCH FILE AND STRIKE ANY KEY WHEN READY
(DISKETTE MIT DER STAPELDATEI EINLEGEN. WENN BEREIT, EINE TASTE BETÄTIGEN)
- A parancs-file-t helyezze a meghajtóba, ha kész, nyomjon le egy billentyűt

DOS. A parancs-file-t tartalmazó lemezt kivettük. A parancsprocesszor a következő parancsot keresi a file-ban. Ha a lemezt a helyes meghajtóba visszatesszük, majd egy tetszőleges billentyűt lenyomunk, folytatódik a feldolgozás.

INSERT DISKETTE FOR DRIVE x AND PRESS ANY KEY WHEN READY:
- helyezzen az x egységbe egy lemezt, majd nyomjon meg egy tetszőleges billentyűt.

DOS. A megjelölt lemezegység logikai egység, s az adott egységben ettől eltérő van. A megfelelő lemez behelyezése után nyomjunk meg egy tetszőleges billentyűt.

INSERT DOS DISK IN D AND STRIKE ANY KEY WHEN READY
(DOS-DISKETTE EINLEGEN IN LAUFWERK D WENN BEREIT, EINE TASTE BETÄTIGEN)
- A DOS lemezt helyezze a meghajtóba, ha kész, nyomjon meg egy billentyűt.

SYS és FORMAT. A FORMAT vagy a SYS megpróbálta a DOS betöltését. A megadott meghajtóban nem DOS rendszerlemez van. Helyezzük be a megfelelő DOS rendszerlemezt, majd nyomjunk meg egy billentyűt.

INSERT DOS DISKETTE IN DRIVE A: PRESS ANY KEY WHEN READY
- A DOS lemezt tegye az A: meghajtóba, ha kész, nyomjon meg egy tetszőleges billentyűt

FDISK. A pillanatnyi fix lemezen a DOS partíciót sikerült előállítani. A DOS lemezt az A: meghajtóba kell tenni, majd egy tetszőleges billentyűt le kell nyomni. Ezután a PC DOS újraindul (BOOT). A DOS a pillanatnyi fix lemezhez azonosítót rendel.

INSERT FIRST DISKETTE IN DRIVE X INSERT SECOND DISKETTE IN DRIVE Y
(ERSTE DISKETTE EINLEGEN IN LAUFWERK X ZWEITE DISKETTE EINLEGEN IN LAUFWERK Y)
- Helyezze az első lemezt az X, a másodikat az Y meghajtóba

DISKCOMP. Az egyik, majd a másik összehasonlítandó lemezt a megadott meghajtókba kell tenni. Ezután a következő üzenet jelenik meg: Press any key when ready (Wenn bereit, eine Taste bet©tigen) ennek jelentése: ha kész, nyomjon le egy billentyűt. Ha ez megtörtént, az összehasonlítás elkezdődik.

INSERT KEYBxx.COM DISKETTE IN DRIVE D: AND STRIKE ANY KEY WHEN READY
- Helyezze be a KEYBxx.COM-ot tartalmazó lemezt az egységbe, majd nyomjon meg egy billentyűt

A rendszer töltésénél jeleztük, hogy a billentyűzet rutin külön lemezen van. Helyezze be a megfelelő lemezt, majd nyomjon meg egy billentyűt.

INSERT LAST BACKUP TARGET IN DRIVE D
- Helyezze be az utolsó mentési cél lemezt a egységbe

BACKUP. A parancs /A paraméterét használtuk. Az utolsó mentési lemez behelyezését kéri a parancs. Nem hibaüzenet.

INSERT NEW DISKETTE FOR DRIVE D AND PRESS ENTER WHEN READY
- Helyezzen be egy új lemezt az egységbe, majd nyomjon meg egy billentyűt. (DOS 3.0-tól <ENTER>-t)

FORMAT. A parancs kéri, hogy a lemezegységbe helyezzünk be egy új lemezt, amit meg akarunk formázni. Ezt követően nyomjuk meg az <ENTER> billentyűt.

INSERT SOURCE DISKETTE IN DRIVE X INSERT TARGET DISKETTE IN DRIVE Y
(QUELLENDISKETTE EINLEGEN IN LAUFTWERK Y)
- Helyezze a forráslemezt az X, a céllemezt az Y meghajtóba

DICKCOPY. A megfelelő lemezeket a megadott meghajtókba kell helyezni, és egy tetszőleges billentyűt le kell nyomni, ezzel folytatódik a másolás.

INSERT TARGET DISKETTE IN DRIVE D: STRIKE ANY KEY WHEN READY
- Helyezze be a céllemezt a meghajtóba, majd nyomjon meg egy billentyűt

DISKCOPY. A parancs akkor használja azt az üzenetet, ha egyetlen hajlékonylemezes egység van csak a rendszerben. Helyezzük be a céllemezt a meghajtóba, majd nyomjunk meg egy tetszőleges billentyűt.

INSERT RESTORE TARGET xx IN DRIVE D: STRIKE ANY KEY WHEN READY
- Helyezze be az egységbe az xx. mentett céllemezt, majd nyomjon meg egy billentyűt

RESTORE. A parancs a következő (xx sosrszámú) céllemezt kéri a d: meghajtóba. A lemez behelyezése után nyomjunk meg egy tetszőleges billentyűt.

INSERT SYSTEM DISK IN D AND STRIKE ANY KEY WHEN READY
- Helyezze be a rendszerlemezt az x egységbe, majd nyomjon meg egy billentyűt

SYS

A parancs megkísérelte betölteni a DOS file-okat, de nem sikerült. Ezért kér a lemezegységbe egy olyan lemezt, amelyik tartalmazza a rendszer-file-okat. A lemez behelyezése után nyomjunk meg egy tetszőleges billentyűt.

INSUFFICIENT DISK SPACE
(NICHT GENÜGEND PLATTENBEREICH)
- Nincs elég hely a lemezen

DOS és PARANCSOK. A lemezen nincs elég szabad hely az adott file felírására. Ebben az esetben a lemez állapotát a CHKDSK-kel ellenőrizhetjük.

INSUFFICIENT MEMORY
(NICHT GENÜGEND SPEICHER)
- Kevés a memória

PARANCSOK. A rendelkezésre álló memóriaterület nem elégséges a parancs-funkció végrehajtására. Ebben az esetben a CONFIG.SYS állományban a BUFFERS = xx értéket

kisebbre kell állítani, majd a rendszert újra kell indítani és a funkciót végre lehet hajtani. Ha ismét ezt az üzenetet kapjuk, akkor a rendszerhez tartozó memória nem elégséges a parancs végrehajtásához.

INSUFFICIENT MEMORY FOR SYSTEM TRANSFER
- Nincs memória a rendszer másolásához

FORMAT vagy SYS. Nincs elég hely a memóriában a rendszer átmásolásához.

INSUFFICIENT ROOM IN ROOT DIRECTORY. ERASE FILES FROM ROOT AND REPEAT CHKDSK
(NICHT GENÜGEND PLATZ IM STAMMVERZEICHNIS / DATEINEN AUS STAMMVERZEICHNIS LÖSCHEN UND CHKDSK WIEDERHOLEN)
- Nincs elég hely a főtartalomjegyzékben, töröljünk a főtartalomjegyzékből file-okat, és ismételjük meg a CHKDSK programot

CHKDSK. A CHKDDSK-kel az elveszített adatblokkokból file-okat lehet újra visszállítani. A főtartalomjegyzék tele van, így esetleg nem mindegyik adatblokk írható fel. Az újra előállítandó file-okat a további ellenőrzés érdekében egy másik lemezre kell másolni, és az átvizsgált lemezről törölni kell. Ezután a maradék elveszett adatokat a CHKDSK-kel újra elő lehet állítani.

INSUFFICIENT SPACE ON DISK
(NICHT GENÜGEND PLATZ AUF PLATTE)
- Nincs elég hely a lemezen

DEBUG. Egy írási parancsot adtunk ki, de a kiírandó adatok számára nincs elég hely a lemezen. Cseréljük ki a lemezt egy olyanra, amely elegendő szabad területtel rendelkezik. Ezután az írási parancsot újból kiadhatjuk. Esetleg töröljünk file-okat a lemezről, de ezután a DEBUG programot ismét be kell tölteni.

INSUFFICIENT STACK SPACE
- Nincs elég hely a veremnek

LINK. Nincs elég hely a programszerkesztő futtatásához. Fel kell szabadítani további memóriarészeket, majd újra futtatni a LINK programot.

INTERMEDIATE FILE ERROR DURING PIPE
(DATAIFEHLER WÄHREND "DATENÜBERGRABE" /PIPE)
- File hiba az átvitel során

DOS. A DOS nem tudta az ideiglenes (temporális) file-okat előállítani, mivel az eredeti tartalomjegyzék tele volt, vagy a DOS az "átviteli" (pipe) állományt nem tudta lokalizálni, vagy a lemezen nincs elég hely az átviendő adatok számára. Ekkor az eredeti tartalomjegyzékből néhány bejegyzést törölni kell, majd a parancsot meg kell ismételni.

INTERNAL STACK OVERFLOW SYSTEM HALTED
- a verem túlcsordult, a rendszer megállt.

DOS. A verem - valószínűleg ismételt hardware megszakItások miatt - megtelt. A CONFIG.SYS file-ban növeljük meg a STACKS értékét.

INVALID BAUD RATE SPECIFIED
(FALSCHE ÜBERTRAGUNGSRATE ANGEGEBEN)
- Helytelen BAUD érték megadás

MODE. Az átviteli sebesség, amelyet paraméterként adtunk meg, nem 100, 150, 300, 600, 1200, 2400, 4800 vagy 9600. (Az érték az első két karakterrel is megadható.)

INVALID CHARACTERS IN VOLUME LABEL
- Érvénytelen karakterek a kötetazonosítóban

FORMAT. A lemezegység címkéjének megadásakor nem megengedet karaktereket írtunk be. A címke megadásakor nem használhatjuk a pontott (.). A címke 1-11 hosszú karaktersorozat.

INVALID CODE PAGE SPECIFIED
- érvénytelen kód-lap.

KEYB. Az adott ország-specifikus információk mellett a kód-lap nem használható.

INVALID COMMAND.COM IN DRIVE D
(UNGÜLTIGER COMMAND.COM IN LAUFWERK x)
- Rossz a COMMAND.COM a meghajtón

DOS. A parancsprocesszor (COMMAND.COM) ismételt betöltése során kiderült, hogy a COMMAND.COM-nak egy érvénytelen verziója került a lemezre. A megfelelő DOS lemezt kell betenni, majd egy tetszőleges billentyű lenyomása után folytatható a munka.

INVALID COUNTRY CODE
- Rossz ország kód

SELECT vagy COUNTRY. A COUNTRY= parancsban vagy a SELECT parancsban megadott országkód nem létezik.

INVALID COUNTRY CODE OR CODE PAGE
- érvénytelen ország kód, vagy kód-lap.

COUNTRY. A CONFIG.SYS file-ban megadott COUNTRY parancs nem megfelelő.

INVALID CURRENT DIRECTORY
- Rossz az aktuális tartalomjegyzék

CHKDSK. A parancs nem képes az aktuális tartalomjegyzék olvasására. Ennek oka a lemez azonosíthatatlan hibája.

INVALID DATE
(UNGÜLTIGES DATUM)
- Helytelen dátum

DATE. A dátumban helytelen értéket vagy elválasztójelet adtunk meg. Érvényes elválasztójelek: a slash (/) és a kötőjel (-). Írjuk be helyesen a napi dátumot!

INVALID DEVICE
(UNGÜLTIGE EINHEIT)
- Helytelen periféria

CTTY. A megadott egységazonosító a DOS számára érvénytelen.

INVALID DEVICE PARAMETERS FROM DEVICE DRIVER
- A periféria-kezelő rossz paramétereket adott meg

FORMAT. A fenntartott szektorok száma nem egész számú többszöröse a szektor/sáv értéknek. Futtassuk újra az FDISK programot, majd kíséreljük meg ismét a FORMAT végrehajtását.

INVALID DIRECTORY
(UNGÜLTIGES INHALTSVERZEICHNIS)
- Helytelen tartalomjegyzék

DOS és PARANCSOK. A megadott úton valamelyik tartalomjegyzék nem létezik.

INVALID DISK CHANGE
- A lemezt nem szabad kicserélni

A nagy kapacitású lemezegységen kicseréltük a lemezt, miközben azon nyitott file-ok voltak. Helyezzük vissza a megfelelő lemezt.

INVALID DRIVE IN SEARCH PATH
(UNGÜLTIGES LAUFWERK IN SUCHPFAD)
- Helytelen meghajtó a keresési úton

DOS. Érvénytelen meghajtó megadás a PATH parancs által megadott úton. A DOS akkor adja ezt az üzenetet, ha parancsot vagy parancsfile-t keres, és nem akkor, amikor a hibás PATH parancsot kiadtuk.

INVALID DRIVE OR FILE NAME
(UNGÜLTIGE LAUFWERKANGABE)
- Helytelen meghajtó specifikáció

DOS, PARANCSOK. A parancsban a meghajtó megadása érvénytelen.

INVALID DRIVE SPECIFICATION. SPECIFIED DRIVE DOS NOT EXIST, OR IS NON-REMOVABLE
- Rossz egység megadása. Az egység nem létezik, vagy fix.

DOS vagy valamelyik parancs. Nem létező lemezegységre hivatkoztunk, vagy fix lemezegységre adtunk ki egy csak cserélhetőre alkalmazható parancsot.

INVALID DRIVE SPECIFICATION. SOURCE AND TARGET DRIVES ARE THE SAME
- Rossz egység megadása. A forrás és a cél lemez ugyanaz.

BACKUP vagy RESTORE. A forrás és cél lemezeket ugyanazon a meghajtón akarjuk használni, ez nem megengedett.

INVALID ENVIRONMENT SIZE SPECIFIED
- Rossz környezet méret

A megadott környezet méret vagy túl nagy, vagy nem-numerikus karaktereket tartalmaz.

INVALID FILE NAME OR FILE NOT FOUND
- Rossz, vagy nem létező file-név

RENAME vagy TYPE. Olyan file-t kíséreltünk meg átnevezni, listázni, amelynek neve rossz volt, vagy nem volt az adott tartalomjegyzékben.

INVALID FORMAT FILE
(UNGÜLTIGE FORMATADATEI)
- Helytelen formájú file

LINK. A könyvtár hibát tartalmaz. Kíséreljük meg újra a szerkesztést.

INVALID KEYBOARD CODE
- Rossz billentyűzet rutin kód

SELECT. Rossz billentyűzet kódot adtunk meg.

INVALID KEYBOARD CODE SPECIFIED
- érvénytelen billentyűzet-kód.

KEYB. A parancsban rossz billentyűzet-kódot adtunk meg.

INVALID MEDIA OR TRACK 0 BAD - DISK UNUSABLE
- Fizikailag nem megfelelő lemez

FORMAT. A parancs nem tudta megformázni a lemezt. Elképzelhető, hogy a lemez 0. sávja fizikailag hibás. Ugyanezt az üzenetet kapjuk, ha a lemez és lemezegység nem kompatibilis. (Pl. egy nagy kapacitású lemezt akarunk megformázni egy 320/360 Kbyte-os egységen.)

INVALID NUMBER OF PARAMETERS
(UNGÜLTIGE ANZAHL PARAMETER)
- Érvényttelen számú paraméter

PARANCSOK. Túl sok, vagy túl kevés paraméter a parancsban.

INVALID NUMERIC PARAMETER
(UNGÜLTIGER NUMERISCHER PARAMETER)
- Helytelen numerikus paraméter

LINK. Egy szám nem-numerikus karaktert tartalmazott.

INVALID NUMERIC SWITCH SPECIFICATION. SWITCH ERROR: "s:xxx"
- Rossz kapcsoló megadása az "s:xxx" kapcsolóban

LINK. A LINK paramétereinek megadása hibás: valamelyik kapcsoló helytelen értéket tartalmaz.

INVALID OBJECT MODULE
(UNGÜLTIGES OBJEKTMODUL)
- Helytelen tárgy-modul

LINK. A tárgy (object) modul hibásan formázott vagy nem teljes. Lehet, hogy a lemez fizikailag hibás.

INVALID PARAMETER
(UNGÜLTIGER PARAMETER)
- Helytelen paraméter

DOS és PARANCSOK. Az adott parancs paraméterei közül egy vagy több érvénytelen. A program olyan meghajtó azonosítót vár, ahol a meghajtó azonosító után kettőspont áll. Meg kell győződni arról, hogy a slash-t (/) követő jel a program (parancs) érvényes paramétere-e.

INVALID PARAMETERS
(UNGÜLTIGE PARAMETER)
- Helytelen paraméterek

MODE. Nem adtunk meg paramétert, vagy az első jel a paraméterben nem L vagy C volt, vagy az első paraméter nem 40, 80, BW40. BW80, CO40, CO80, L, R volt, vagy a képernyő adapter (színes/grafikus), amire a parancs vonatkozik, nincs a rendszerben.

INVALID PARTITON TABLE
(UNGÜLTIGE PARTITONTABELLE)
- Helytelen partíció tábla

START. A DOS-t fix lemezről indítottuk, a lemez partíció táblázataiban érvénytelen információk fordulnak elő. Ekkor a DOS-t a floppy lemezről kell indítani, majd az FDISK segítségével korrigálni kell a fix lemez partíció-információit.

INVALID PATH
(UNGÜLTIGER PFAD)
- Helytelen útmegadás

Valamelyik parancsban megadott út rossz. Vagy nem megengedett karaktereket tartalmaz, vagy hosszabb 63 karakternél.

INVALID PATH, NOT DIRECTORY OR DIRECTORY NOT EMPTY
(UNGÜLTIGER PFAD, KEIN INHALTSVERZEICHNIS ODER INHALTSVERZEINCHNIS NICHT LEER)
- Helytelen útmegadás, nem tartalomjegyzék vagy nem üres a tartalomjegyzék

RMDIR. A megadott tartalomjegyzéket nem lehet törölni, mivel az útmegadásban levő nevek érvénytelenek voltak, vagy a megadott tartalomjegyzék tartalmaz még más tartalomjegyzékre vonatkozó altartalomjegyzék bejegyzéseket, vagy file azonosítókat. A tartalomjegyzék csak akkor törölhető, ha üres. (Kivétel a . és a .. bejegyzés.)

INVALID PATH OR FILE NAME
(UNGÜLTIGER PFAD ODER DATEINAME)
- Érvénytelen út vagy file azonosító

COPY vagy ATTRIB. Nem létező file-azonosítót vagy tartalomjegyzéket adtunk meg.

INVALID PATH OR PARAMETER
- érvénytelen út, vagy paraméter.

APPEND. A parancsban a /X, /E kapcsolókat útleírásukkal együtt használtuk, ami nem megengedett.

INVALID SIGNATURE IN COUNTRY.SYS FILE
- érvénytelen fejléc a COUNTRY.SYS file-ban.

SELECT. A COUNTRY.SYS file-ban a SELECT érvénytelen fejlécet talált. Ellenőrizzük a file-t, majd ismételjük meg a parancsot.

INVALID SYNTAX ON PRINTER.SYS CODE PAGE SWITCHING DEVICE DRIVERS
- nem megfelelő szintaxis a PRINTER.SYS kód-lap-váltó használatakor.

MODE. A PRINTER.SYS-t tartalmazó DEVICE konfigurációs parancs paraméterei nem megfelelőek. Javítsuk ki a CONFIG.SYS file-t, majd indítsuk újra a DOS-t.

INVALID SUBDIRECTORY
(UNGÜLTIGES UNTERVERZEICHNIS)
- Helytelen altartalomjegyzék

CHKDSK. A megadott altartalomjegyzékben érvénytelen információk találhatók. A CHKDSK megkísérli a hibát átlépni. A hiba típusáról akkor kapunk információt, ha a /V paraméterrel hívjuk meg a CHKDSK programot.

INVALID SWITCH
(UNGÜLTIGER SCHALTER)
- Helytelen kapcsoló

LINK. A kijelzett karakter nem érvényes LINK paraméter kapcsoló.

INVALID SWITCH CHARACTER
- Rossz karakter a kapcsolóban

VDISK. A konfigurációs parancs egy slash-t (/) talált, de az ezt követő karakter nem E volt. A memória lemezt a parancs a memória alsó részében hozta létre.

INVALID TIME
(UNGÜLTIGE ZEIT)
- Érvénytelen idő megadás

TIME. Érvénytelen értéket vagy elválasztó jelet adtunk meg. Az órák, percek, másodpercek között csak kettőspont (:), a másodpercek és a századmásodpercek között a pont (.) az érvényes elválasztó jel.

INVALID VOLUME LABEL
- Rossz kötetazonosító

FORMAT. A kért címke (kötetazonosító) nem egyezett a lemezegység címkéjével. A VOL kiadásával ellenőrizhető, mi is a helyes név. Ezután újra ki lehet adni a FORMAT parancsot.

IS KEYBxx.COM ON ANOTHER DISKETTE (Y/N)?
- A KEYBxx.COM másik lemezen van (I/N)?

A KEYBxx.COM file-t nem találta a rendszer, ezért megkérdezte, hogy másik lemezen van-e. Ha rossz billentyűzet rutin kódot adtunk meg, akkor N-nel kell válaszolni, majd beírni a helyes kodot. Ha külön lemezen tartjuk, akkor Y a helyes válasz.

K...

KEYB HAS NOT BEEN INSTALLED
- a KEYB programot még nem installáltuk.

KEYB. A KEYB parancs lekérdező funkcióját akartuk használni, holott még nem is installáltuk. Töltsük be a KEYB-t, majd ismételjük meg a parancsot.

KEYBOARD ROUTINE NOT FOUND
- A billentyűzet rutin nem található meg

A billentyűzet rutint nem találta a rendszer sem az aktuális tartalomjegyzékben, sem külön lemezen.

L...

LABEL NOT FOUND
(SPRUNGZIEL NICHT GEFUNDEN)
- Az ugrási címke nem található

PARANCSFILE-OK. A GOTO parancsban olyan címke van megadva, amely a parancsfile-ban nem található.

LAST BACKUP TARGET NOT INSERTED
- Nem az utolsó mentési lemezt tettük be

BACKUP. A parancsban használtuk a /A paramétert, de a cserélhető céllemez nem az utolsó volt a mentési lemezek közül. Tegyük be az utolsó mentési lemezt.

***** LAST FILE NOT BACKED UP *****
- Az utolsó file nincs elmentve

BACKUP. A parancs fix lemezegységre mentett. Az utoljára kimentett file másolása közben megtelt a lemez. Ezért ezt a file-t a mentésből törli a parancs. A mentés befejeződik.

LINE TOO LONG
(ZEILE ZU LANG)
- A sor túl hosszú

EDLIN. Egy jelsorozat kicserélése során a 253 sorhosszúságot átléptük. A "szöveg-rész kicserélése" parancs nem hajtódik végre. A sort rövidebb jelsorozatokra kell felbontani, majd a parancsot ismét ki kell adni.

LIST OUTPUT IS NOT ASSIGNED TO A DEVICE
(LISTAUSGABE IST KEINER EINHEIT ZUGEORDNET)
- A lista output nincs egységhez hozzárendelve

PRINT. A PRINT listázó egysége nem érvényes egység. A PRINT parancsot ismét ki kell adni, de érvényes nevet kell megadni.

LOCK VIOLATION
- Zárolási hiba

XCOPY. A forrás file, vagy annak egy része zárolt. Várni kell egy kis ideig, majd újból kiadni a parancsot.

LOGGING TO FILE <file-név>
- bejegyzések a <file-név> file-ba.

BACKUP. Információs üzenet: a /L paraméterrel megadott <file-név> file-t használja a parancs LOG-file-ként.

LOGICAL DOS DRIVE CREATED,
DRIVE LETTERS CHANGED OR ADDED
- új logikai DOS egység, az egységek betűjele módosítva, vagy bővítve.

FDISK. A kiterjesztett DOS partícióban új logikai egységet hoztunk létre. Ennek hatására a már meglévő egységekhez tartozó betűjelek módosulhatnak, s legalább egy további betű is 'érvényes' lesz.

INVALID SIGNATURE IN KEYBOARD.SYS FILE
- érvénytelen fejléc a KEYBOARD.SYS file-ban.

SELECT. A KEYBOARD.SYS file-ban a SELECT érvénytelen fejlécet talált. Ellenőrizzük a file-t, majd ismételjük meg a SELECT parancsot.

INVALID STACK PARAMETER
- érvénytelen STACK paraméter.

STACKS. A CONFIG.SYS file-ban szereplő konfigurációs parancs paramétereit rosszul adtuk meg

LPT#: NOT REROUTED
(LPT#: NICHT UMGELEITET)
- LPT#: nem átirányítható

MODE. A párhuzamos interface-szel rendélkező nyomtató outputját egy soros egységre átirányítottuk. A MODE paranccsal viszont a nyomtatás szélességét vagy sűrűségét változtattuk. A rendszer az átirányítást érvényteleníti.

LPT#: REROUTED TO COMn
(LPT#: UMGELEITET ZU COMn)
- LPT#: átirányítva a COMn-re

MODE. A párhuzamos interface-szel rendelkező nyomtató outputját egy soros egységre átirányítottuk. (# = 1, 2 vagy 3; n = 1 vagy 2)

LPT#: SET FOR 80
(LPT#: AUF (80 GESETZT)
- LPT érték 80

MODE. A nyomtatási szélességet úgy próbáltuk 80-ra állítani, hogy a standard formátumot követeljük meg. Ha ez nem sikeres, akkor ezt az üzenetet egy hibaüzenet követi a képernyőn. (# = 1, 2 vagy 3)

LPT#: SET FOR 132
(LPT#: AUF 132 GESETZT)
- LPT#: érték 132

MODE. A nyomtatási szélességet úgy próbáltuk 132-re állítani, hogy a sűrített nyomtatási formát követeljük meg. Ha ez nem sikeres, akkor egy hibaüzenet is megjelenik a képernyőn. (# = 1, 2 vagy 3)

M...

MAKE SURE A DISKETTE IS INSERTED INTO THE DRIVE AND THE DOOR IS CLOSED
- Győződjön meg arról, hogy van-e lemez az egységben, és az ajtó zárva van-e

DISKCOMP vagy DISKCOPY. A hajlékonylemezes egység üres, vagy nincs az ajtaja becsukva.

MAXIMUM AVAILABLE SPACE IS NNNN CYLINDERS AT CYLINDER XXXX
MAXIMUM AVALLABLE SPACE FOR PARTITION IS xxx CYLINDERS
(MAXIMAL VERFÜGBARER BEREICH IST NNNN ZYLINDER BEI ZYLINDER XXXX)
- Maximális rendelkezésre álló terület az xxxx-től nnnn darab cilinder

FDISK. A "DOS partíció előállítása" lehetőséget kértük. A DOS partíció elhelyezkedését adjuk meg ezzel. Ezeket az értékeket standardként is használhatjuk.

MAXIMUM NUMBER OF LOGICAL DOS DRIVERS INSTALLED
- maximális számú logikai egységet már installáltunk.

FDISK vagy DOS. A DOS logikai egységei A-tól Z-ig terjedő betűjelet kapnak, így legfeljebb 24 logikai egység lehet. Ennél többet próbáltunk meglétrehozni. Ha a 25.-re is szükség van, azt csak úgy tehetjük meg, hogy valamelyik előzőt töröljük.

MEMORY ALLOCATION ERROR. CANNOT LOAD COMMAND, SYSTEM HALTED
(FEHLER BEI DER SPECHERZUORDNUNG. COMMAND KANN NICHT GELADEN WERDEN, SYSTEM ANGEHALTEN)
- Memória lefoglalási hiba miatt a parancsprocesszort nem lehet betölteni, a rendszer leállt

DOS. Egy program felülírta azt a memóriaterületet, amelyen a DOS allokált memóriablokkok láncát tartja nyilván. A DOS-t ismét be kell tölteni. (boot)

MISMATCH DOS LEVEL NUMBER
- Eltérő DOS szint szám

LINK. Programhiba. A program szállítójával kell felvenni a kapcsolatot, és tisztázni a hiba okát.

MISSING FILE NAME
(FEHLENDER DATEINAME)
- Hiányzó file név

RENAME. A második file-nevet nem adtuk meg.

MISSING FROM THE FILE IS EITHER THE DEVICE ID OT THE CODE PAGE
- a file-ból vagy az egység vagy a kód-lap hiányzik.

MODE. A kód-lapok listáján megadott sorszámok vagy a használt nyomtatótípusának megfelelő információ hiányzik a CPI file-ból.

MISSING OPERATING SYSTEM
(FEHLENDES BETRIEBSSYSTEM)
- Nincs operációs rendszer

START. A DOS fix lemezes indítása során a betöltő eljárás megállapította, hogy a DOS partíció indítóképesként (aktív) lett megjelölve, ennek ellenére DOS rendszert nem tartalmaz. A DOS rendszert a fix lemezre kell másolni. (A FORMAT parancsot a /S paraméterrel kell megadni, majd a rendszer file-okat is a fix lemezre kell másolni.)

MODE fff CODE PAGE FUNCTION COMPLETED
- MODE: az fff kód-lapozási funkció végrehajtva.

MODE. Információs üzenet. A QUERY, PREPARE vagy SELECT alparancsok (ez fff értéke) végrehajtása befejeződött.

MUST SPECIFY COM1 OR COM2
- COM1 vagy COM2 megadása kötelező

MODE. A parancsban nem adtuk meg a COM1 vagy COM2 használatát.

MUST SPECIFY ON OR OFF
- Csak ON vagy OFF adható meg

BREAK. A parancsnak ket paramétere lehet: ON vagy OFF. Ettől eltérő paramétert adtunk meg.

MUST SPECIFY DESTINATION LINE NUMBER
(NUMMER DER ZIELZEILE IST ANZUGEBEN)
- Meg kell adni a célsorok számát

EDLIN. A MOVE vagy a COPY parancsot sorszám-azonosító nélkül adtuk ki. A parancsot helyesen ismételten ki kell adni.

MUST SPECIFY ON OR OFF
- csak ON vagy OFF adható meg.

BREAK. A parancsban az ON vagy OFF szócskától eltérő paramétert adtunk meg.

N...

NAME OF LIST DEVICE [PRN]:
(NAME DER LISTENEINHEIT [PRN]:)
- A listázó egység azonosítója [PRN]:

PRINT. Ezt a nyomtatási folyamat indítása után kapjuk. Erre egy fenntartott egység azonosítójával, vagy az üzenetvég billentyű lenyomásával kell válaszolni, és ekkor az első párhuzamos interface-szel rendelkező nyomtatóra kerül a lista.

NEAR/FAR CONFLICT
- Közeli/távoli megadási hiba

LINK. Egy COMMON változó közeli (near) és távoli (far) attributum definíciója ellentmondó. Javítsuk ki, és futtassuk újra a LINK programot.

NETWORK SUPPORT LOADED. UNABLE TO FDISK
- FDISK hálózaton nem használható

FDISK. Egy hálózaton implementált fix lemezegységre nem használhatjuk az FDISK programot, még akkor sem, ha fizikailag ugyanazon a gépen van.

NLSFUNC ALREADY INSTALLED
- az NLSFUNC programot már installáltuk.

NLSFUNC. A parancs rezidens része már a tárban van; másodszor nem töltődik be.

NO APPEND
- nincs 'APPEND' hatálya alatt semmi.

APPEND. Az APPEND parancs lekérdező funkcióját használtuk. Válaszként megtudtuk, hogy nincs egyetlen olyan út sem, amelyiket az APPEND hatására a DOS átnézne. Az APPEND; kiadása után pl. ezt a választ kapjuk.

NO CODE PAGE HAS BEEN SELECTED
- nincs kiválasztott kód-lap.

MODE. A STATUS alparancsra válaszként kapjuk ezt az üzenetet. Jelentése a szóbanforgó eszközre egyetlen kód-lapot sem választottunk ki.

NO EXTENDED DOS PARTITION TO DELETE
- nincs kiterjesztett DOS partíció, ezértnem törölhetjük.

FDISK. Az FDISK parancsot kiterjesztett DOS partíció törlésére utasítottuk, holott az nem létezik, s ezért nem is tudjuk törölni.

NO DOS PARTITION TO DELETE
(KEINE DOS PARTITON ZU LÖSCHEN)
- Nincs törölhető partíció

FDISK. A felhasználó a "DOS partíció törlése" lehetőséget választotta, de az adott fix lemezen nincs DOS partíció.

NO FILES ADDED
- Nem adott hozzá file-t

REPLACE. A /A paramétert használtuk, de a forrás file-ok már mind léteznek a cél lemezen is.

NO FILES FOUND
- Nem talált egy file-t sem

REPLACE. A forrásban kért file-ok egyikét sem találta meg a parancs a cél lemezen.

NO FILES REPLACED
- Nincs mit helyettesíteni

REPLACE. A cél lemezen egyetlen file-t sem helyettesített a parancs, mert a forrás file-ok mindegyikére azt mondtuk, hogy nem kell helyettesíteni.

NO FIXED DISKS PRESENT
(KEINE FESTPLATTE VORHANDEN)
- A fix lemezt a rendszer nem éri el

FDISK. Az FDISK programot egy olyan PC-n hajtjuk végre, amelynek:
- nincs fix lemeze;
- a fix lemez a bővítési egységben tallálható, de az nincs bekapcsolva;
- a fix lemezt rosszul installálták

NO FREE FILE HANDLERS. CANNOT START COMMAND, EXITING
(KEINE FREIEN DATEINUMMERN. COMMAND KANN NICHT GESTARTED WERDEN, BEENDET)
- Nincs szabad file-kezelő. A START nem hajtható végre, kilépés

DOS. A parancsprocesszort (COMMAND.COM) nem lehetett ismételten behívni, mivel pillanatnyilag túl sok file van megnyitva. A CONFIG.SYS adatállományban lévő FILES = xx paraméterben a file-ok számát növelni kell, majd a DOS-t újra kell indítani.

NO LOGICAL DRIVES DEFINED
- nincs logikai egység definiálva.

FDISK. A kiterjesztett DOS partícióban egyetlen logikai egységet sem hoztunk létre. Ha az allokált lemez-területet egyáltalán használni akarjuk, akkor legalább egy logikai egységet létre kell hoznunk a kiterjesztett DOS partícióban.

NO OBJECT MODULES SPECIFIED
(KEINE OBJEKTMODULE ANGEGEBEN)
- Nem adtunk meg object modult

LINK. A parancssorban nem adtunk meg tárgy (object) modult, a LINK-nek viszont szüksége van rá.

NO PARTITIONS DEFINED
- nincs egyetlen partíció sem.

FDISK. A program jelzi, hogy a fix lemezen még egyetlen partíciót sem hoztunk létre. Ha a lemezt használni szeretnénk, legalább egyet létre kell hozni.

NO PARTITIONS TO DELETE
- nincs a lemezen partíció, ezért nem is törölhetjük azokat.

FDISK. A programot partíció törlésére utasítottuk, holott nincs egyetlen partíció sem.

NO PARTITONS TO MAKE ACTIVE
(KEINE PARTITONEN ZUM AKTIVIEREN)
- Nincs aktivizálható partíció

FDISK. A felhasználó az "aktív partíció változtatása" lehetőséget választotta, de az adott fix lemezen nincs ilyen. Ezért először a "DOS partíció előállítása" lehetőséget kell választani, és csak ezután aaktivizálható ez a partíció.

NO PATH
(KEIN PFAD)
- Nincs út

PATH. Pillanatnyilag a DOS számára nincs alternatív út kijelölve, amelyen a parancsokat vagy parancs file-okat kereshetné. Ezt az üzenetet akkor kapjuk, ha azokat a megadott vagy az aktuális tartalomjegyzékben nem találja.

NO PRIMARY DOS PARTITION TO DELETE
- nincs elsődleges DOS partíció, ezért azt nem is törölhetjük.

FDISK. A programot az elsődleges DOS partíció törlésére utasítottuk, holott még létre sem hoztuk.

NO RETRY ON PARALLEL PRINTER TIME-OUT
- Időtúllépés esetén nincs ismétlés

MODE. Az üzenet tájékoztat arról, hogy a P paramétert nem adtuk meg, ezért az időtúllépés esetén a nyomtatás megszakad, és nem kísérletezik tovább.

NO ROOM FOR SYSTEM ON DESTINATION DISK
(KEIN PLATZ FÜR DAS SYSTEM AUF DER ZIELDISKETTE)
- Nincs hely a rendszer számára a céllemezen

SYS. A feltett lemezen nem található a DOS számára fenntartott terület, ezért a rendszer nem vihető át. Ekkor egy üres lemezt meg kell formázni (FORMAT /S), és így lehet a rendszert létrehozni.

NO ROOM IN DIRECTORY FOR FILE
(DAS INHALTSVERZEICHNIS IST VOLL)
- Nincs hely a tartalomjegyzékben a file számára

EDLIN. A tartalomjegyzék a megadott lemezen megtelt. A végrehajtott változtatások elvesztek. Indítsuk újra az EDLIN-t egy üres, formázott lemezre.

NO ROOM IN ROOT DIRECTORY
- Nincs hely a főtartalomjegyzékben

LABEL. A címke (kötetazonosító) létrehozása a főtartalomjegyzékben egy bejegyzést igényel. Ez azonban nem lehetséges, mert az tele van.

NO SOURCE DRIVE SPECIFIED
- a forrás egységet nem adtuk meg.

BACKUP. A forrás egységet meg kell adni.

NO SPACE FOR A XXXX CYLINDER PARTITON
(KEIN PLATZ FÜR EINE PARTITON IT XXXX ZYLINDERN)
- Nincs hely az xxxx cilinderes partíció számára

FDISK. A megadott partíció cilinderméret nagyobb, mint a szabad terület a lemezen. Ebben az esetben kisebb értéket kell megadni.

NO SPACE FOR A NNNN CYLINDER PARTITION AT CYLINDER XXXX
(KEIN PLATZ FÜR EINE PARTITION MIT NNNN ZYLINDERN BEI ZYLINDER XXXX)
- Az xxxx cilindernél lévő nnnn cilinderes partíció számára nincs hely

FDISK. Egy partíciót a fix lemez egy meghatározott helyén akartunk előállítani. Ezen a helyen viszont nincs elég szabad terület a DOS partíció számára.

NO SPACE TO CREATE A DOS PARTITON
(KEIN PLATZ FÜR DAS ERSTELLEN EINER DOS-PARTITON)
- Nincs hely a DOS partíció létrehozásához

FDISK. A "DOS partíció előállítása" lehetőséget választottuk. A fix lemezen viszont nincs több szabad hely.

NO SPACE TO CREATE LOGICAL DRIVE
- nincs hely a logikai egység létrehozására.

FDISK. A kiterjesztett DOS partícióban levő összes helyet már más logikai egységek használják, ezért továbbiakat már nem hozhatunk létre.

NO SUB-DIRECTORIES EXIST
(KEINE UNTERVERZEICHNISSE VORHANDEN)
- Altartalomjegyzékek nem léteznek

TREE. A megadott meghajtón csak eredeti tartalomjegyzék van, ezért elérési utat nem kell megadni.

NO SYSTEM ON DEFAULT DRIVE
- Az aktuális lemezen nincs rendszer

SYS. Az átmásolandó rendszer file-okat a parancs nem találta az aktuális lemezegységen.

NO TARGET DRIVE SPECIFIED
- nem adtuk meg a cél egységet.

BACKUP. A parancsban mind a forrás, mind a cél egységet meg kell adni.

NO VERSION OF GRAPHIC CHARACTER SET TABLE IS ALREADY LOADED
- a grafikus karakterkészlet semelyik verziója sincs betöltve.

GRAFTABL. A parancs jelzi, hogy nem töltötte be a grafikus karakterkészletet, mert a DOS már betöltötte. Információs üzenet.

NON-DOS DISKETTE
- Nem DOS lemez

CHKDSK vagy valamelyik parancs. A lemez formátuma felismerhetetlen.

NON-STANDARD VERSION OF GRAPHIC CHARACTER SET.TABLE IS ALREADY LOADED

GRAFTABL. Az üzenet jelzi, hogy a memóriába töltött grafikus karakterkészletet módosítottuk, ezért azt a rendszer nem tudja azonosítani. Ha a GRAFTABL segítségével egy új készletet töltünk be, akkor az a RAM-ban levő táblázatot felülírja.

NON-SYSTEM DISC OR DISC ERROR. REPLACE AND STRIKE ANY KEY WHEN READY
(KEINE-SYSTEMPLATTE ODER PLATTENFEHLER. AUSTAUSCHEN WENN BEREIT, EINE TASTE BETÄTIGEN)
- Nem rendszerlemez vagy rendszerhiba. Cseréljük ki a lemezt, majd nyomjunk le egy tetszőleges billentyűt.

START. Az xxxBIO.COM vagy az xxxDOS.COM számára nincs bejegyzés a tartalomjegyzékben, vagy a rendszer indítása során olvasási hiba fordult elő. Az A: meghajtóba DOS lemezt kell helyezni, majd a rendszert újraindítani.

***** NOT ABLE TO BACK UP FILE *****
- A file nem menthető

BACKUP. A file-t nem lehet menteni, mert osztott. Próbáljuk meg később a mentést, és használjuk a /M paramétert.

***** NOT ABLE TO RESTORE FILE *****
- A file nem tölthető vissza

RESTORE. A visszatöltendő file-t nem lehet megnyitni, mert zárolt.

NOT ENOUGH MEMORY
- Nincs elég memória

SHARE vagy REDIR. A felhasználható memória kevesebb, mint amennyi a konfigurációs parancsok végrehajtásához szükséges, ezért ezek installálására nem került sor.

NOT ENOUGH ROOM TO MERGE THE ENTIRE FILE
(NICHT GENÜGEND PLATZ ZUM EINLESENDER GESAMTEN DATEI)
- Nincs elég terület a teljes file szerkesztéséhez

EDLIN. Az átviteli parancs a megadott file teljes tartalmát nem tudta beolvasni a memóriába, mivel nem állt rendelkezésére elég szabad terület. A file-nak csak egy részét olvasta be.

NOT FOUND
(NICHT GEFOUND)
- Nem található

EDLIN. A megadott sor-intervallum nem tartalmazza azt a jelsorozatot, amit a "szöveg keresése" parancsban megadtunk. Előfordulhat például, hogy az OK üzenetre N-nel válaszoltunk, de a megadott jelsorozat többször már nem fordult elő a szövegben.

O...

ONE OR MORE CON CODE PAGES INVALID FOR GIVEN LANGUAGE
- néhány CON kód-lap érvénytelen az adott nyelv esetén.

KEYB. A KEYB parancs a használt kód-lapokhoz további táblázatokat hoz létre. Ha valamelyik kód-lap sorszáma érvénytelen az adott nyelv esetén, akkor ezt nem tudja létrehozni. Az érvényes kód-lapokban ezt meg tudja tenni, azok használhatók.

ONLY NON-BOOTABLE PARTITIONS EXIST
- csak nem tölthető partíciók léteznek.

FDISK. A programot az aktív partíció módosítására utasítottuk, holott a fix lemezen egyáltalán nem volt aktív partíció.

ONLY PARTITIONS ON DRIVE 1 CAN BE MADE ACTIVE
- csak az első fix egységen hozható létre aktív partíció.

FDISK. Operációs rendszer csak az első fix lemezről tölthető. Ezért a többi fix egységen nem hozhatunk létre aktív partíciót.

OUT OF ENVIRONMENT SPACE
- nincs elég hely a program-környezetben.

DOS. A rendszer nem tudta végrehajtani a SET parancsot, mert a program-környezetben már nem volt elég hely.

OUT OF ENVIRONMENT SPACE
(KEIN UMGEBUNSBEREICH MEHR VORHANDEN)
- Nincs több környezet terület

DOS. A DOS nem fogadta el a SET parancsot, mivel azt a területet nem tudta kibővíteni, Ez akkor fordulhat elő, ha egy rezidens program betöltése után újabb információkat akarunk a környezethez hozzáfűzni.(Rezidens pl. egy egyszerű billentyűzet rutin, PRINT, MODE vagy GRAPHICS parancs.)

OUT OF SPACE ON LIST FILE
(KEIN PLATZ MEHR IN LISTDATEI)
- A lista-file-ban nincs több hely

LINK. Ez a hiba akkor lép fel, ha a lemezen nincs több hely a lista file számára.

OUT OF SPACE ON RUN FILE
(KEIN PLATZ MEHR IN AUSFÜHRUNGSDATEI)
- Nincs több hely a végrehajtási file-ban

LINK. Ez a hiba akkor lép fel, ha a lemezen nincs elég hely a végrehajtási file (.EXE) számára.

OUT OF SPACE ON SCRATCH FILE

LINK. Az a lemez, amelyiken a programszerkesztő a VM.TMP (scratch) file-t letre akarja hozni, tele van.

OUT OF SPACE ON VM.TMP
(KEIN PLATZ MEHR IN VM.TMP)
- A VM.TMP-ben nincs több hely

LINK. A lemezen nincs több hely a VM.TMP file bővítésére.

P...

PARAMETERS NOT COMPATIBLE
(PARAMETER NICHT VERTRÄGLICH)
- A paraméter nem használható ebben az alakban.

Az üzenetet több parancs is küldheti. A megadott paraméterek kombinációja nem megfelelő.

PARAMETER NOT COMPATIBLE WITH FIXED DISK
- a paraméter nem használható fix lemezre.

FORMAT. A parancs paraméterben /1-et vagy /8-at adunk meg, miközben a fix lemezt formáztuk. Ezek a paraméterek fix lemezre nem érvényesek.

PARITY ERROR OR NONEXISTENT MEMORY ERROR DETECTED

DEBUG. A program a memória írása/olvasása közben paritáshibát észlelt. Ennek oka lehet az is, hogy nemlétező memóriacímet használtunk. Ha a hiba ismétlődik, jelentsük a gép szállítójának is.

PARTITION SELECTED (#) IS NOT BOOTABLE, ACTIVE PARTITION NOT CHANGED
- a kiválasztott partíció nem tölthető, az aktív partíció nem változott.

FDISK. Aktív partíció csak tölthető lehet. Ha nem ilyen partícióról van szó, azt nem tudjuk aktív partíciónak kijelölni.

PARTITON 1 IS ALREADY ACTIVE
- Az 1 partíció már aktív

FDISK. Tájékoztató üzenet: a fix lemezen egyetlen partíció van, s az már aktív.

PARTITION xx MADE ACTIVE
- Az xx partíció aktív

FDISK. Tájékoztató üzenet: a parancs az xx partíciót aktivizálta. A rendszer újraindításkor innen töltődik be.

PATH NOT FOUND
- Az út nem létezik

Valamelyik parancsban olyan utat adtunk meg, amelyik nem létezik az adott lemezegységen.

PATH TOO LONG
- Az út túl hosszú

Valamelyik parancsban 63 karakternél hosszabb utat adtunk meg.

PATHNAME TOO LONG
- Az útnév túl hosszú

PRINT

Valamelyik nyomtatandó file-hoz tartozó elérési út több mint 63 karakterből áll.

PLEASE REPLACE ORIGINAL DISKETTE IN DRIVE A: AND PRESS <ENTER>
- Helyezze vissza az eredeti lemezt az A: egységbe, majd nyomja meg az <ENTER> billentyűt

LINK. Az üzenetet a pogram az .EXE file írása előtt adja ki, feltéve, hogy a /P kulcsot megadtuk. Helyezzük be a lista file-t tartalmazó lemezt, és nyomjuk le az <ENTER> billentyűt.

PRESS ANY KEY TO BEGIN ADDING FILES
- A file-ok hozzáadásához nyomjon meg egy billentyűt.

REPLACE. Az /A /N paramétereket egyszerre adtuk meg. Ha kész vagyunk, akkor nyomjunk meg egy tetszőleges billentyűt.

PRESS ANY KEY TO BEGIN COPYING FILE(S)
- A másolás megkezdéséhez nyomjon meg egy billentyűt

XCOPY. A másolás megkezdése előtt a program megáll, hogy legyen idő a lemezek cseréjére. TetszÖleges billentyű lenyomására a másolás megkezdődik.

PRESS ANY KEY TO BEGIN FORMATTING d:
(EINE TASTE BETÄTIGEN - START FORMATIEREN d:)
- Ha lenyom egy billentyűt, az egységen elkezdődik a formázás

FORMAT. A fix lemezt (d: meghajtón) kell formázni. A formázás során valamennyi adat elvész a lemezről. A <CTRL+<BREAK> lenyomásával nem indul, más billentyű lenyomása esetén elkezdődik a formázás.

PRESS ANY KEY TO BEGIN RECOVERY OF THE FILE(S) ON DRIVE x
- a file-ok visszaállításához nyomjon meg egy billentyűt.

RECOVER. A megfelelő egységben helyezzük be azt a lemezt, amelyen a file-okat vissza akarjuk állítani, majd nyomjunk meg egy tetszőleges billentyűt.

PRESS ANY KEY TO BEGIN REPLACING FILE(S)
- A file-ok helyettesítéséhez nyomjon meg egy billentyűt

REPLACE. A parancs végrehajtása megállt, hogy ki lehessen cserélni a lemezt. Tetszőleges billentyű megnyomására folytatja a parancs végrehajtását.

PREVIOUSLY PREPARED CODE PAGE REPLACED
- az előzőleg előkészített kód-lapot kicseréltem.

MODE. A kód-lapok tárolására szolgáló hely már megtelt, ezért a PREPARE alparancs valamelyik előzőleg előkészített kód-lapot lecserélte az újra. A MODE STATUS segítségével megtudhatjuk, hogy mely kód-lapok maradtak érvényesek.

PRIMARY DOS PARTITION ALREADY EXISTS
- elsődleges DOS partíció már létezik.

FDISK. Újabb elsődleges DOS partíciót akartunk a fix lemezen létrehozni, holott csak egy lehet.

PRIMARY DOS PARTITION CREATED
- elsődleges DOS partíció létrehozva.

FDISK. A program a fix lemezen létrehozta az elsődleges DOS partíciót.

PRIMARY DOS PARTITION DELETED
- elsődleges DOS partíció törölve.

FDISK. A program törölte az elsődleges DOS partíciót.

PRINT QUEUE IS EMPTY
(WARTESCHLANGE IST LEER)
- A PRINT sor üres

PRINT. A PRINT nem dolgoz fel pillanatnyilag file-okat.

PRINT QUEUE IS FULL
(WARTESCHLANGE IST VOLL)
- A PRINT sor megtelt

PRINT. A várakozási sorba több mint 10 file-t akartunk besorolni. A várakozási sorba egy újabb file csak akkor léptethető be, ha egy nyomtatás befejeződött., vagy a sorból file-t töröltünk.

PRINTER ERROR
(DRUCKERFEHLER)
- Nyomtató hiba

MODE. A MODE parancs nem tudta a nyomtatási módot beállítani. Okai: I/O hiba; a papír kifogyott (nyomtató kikapcsolva); idő túllépés (nincs készenlétben).

PRINTER LINES PER INCH SET
(SETZEN DER DRUCKERZEILEN PRO ZOLL)
- A nyomtatott sorok száma inch-enként beállítva

MODE. Megkíséreltük a sortávolságot 6 vagy 8 inch-re állítani. Ha ez sikertelen volt, akkor ezt egy hibaüzenet követi a képernyőn.

PROBABLE NON-DOS DISK, CONTINUE (Y/N)?
(WAHRSCHEINLICH KEINE-DOS-DISKETTE, FORTFAHREN (J/N)?)
- Valószínűleg nem DOS lemez. Folytassuk (I/N)?

CHKDSK. A file-leíró táblázat (FAT) azonosító byte-jai érvénytelen információt tartalmaznak. A lemezt vagy nem DOS alatt formázták, vagy súlyosan megsérült. Ha a /F paramétert nem használtuk és Y-vel, ill J-vel felelünk, akkor a CHKDSK egy lehetséges hibaelhárítási megoldást ad anélkül, hogy a lemezt valójában változtatná.

PROCESSING CANNOT CONTINUE
(VERARBEITUNG KANN NICHT FORTGESETZT WERDEN)
- A feldolgozás nem folytatható

CHKDSK. Ezt egy további üzenet is követi, amely megadja, hogy a CHKDSK miért nem folytatható. Ennek általában kevés memória az oka.

PROGRAM SIZE EXCEEDS CAPACITY OF LINK
(PROGRAMMGRÖBE ÜBERSCHREITET KAPAZITÄT VON LINK)
- A program mérete túllépi a LINK címtartományát

LINK. A modul, amit a LINK szerkeszt, túl nagy.

PROGRAM TERMINATED NORMALLY
- A program rendben lefutott

DEBUG. Tájékoztató üzenet. A G(o), T(race) vagy P(roceed) paranccsal elindított program INT 20, INT 27, INT 21, vagy egyéb támogatott módon állt meg.

PROGRAM TOO BIG TO FIT IN MEMORY
(PROGRAMM FÜR DEN SPEICHER ZU GROB)
- A program nem fér a memóriába

DOS. A külső (external) típusokat tartalmazó file-ok nem tölthetők be, mivel méretük nagyobb, mint a rendelkezésre álló memória. Ebben az esetben a CONFIG.SYS file-ban a BUFFERS = xx értéket csökkenteni kell, majd a rendszert újra kell indítani. Ezután a parancsot újra ki lehet adni. Ha ismét ezt az üzenetet kapjuk, akkor a rendszer nem rendelkezik a parancs végrehajtásához szükséges memóriával.

R...

READ ERROR, COUNTRY.SYS
READ ERROR, KEYBOARD.SYS
- a COUNTRY.SYS(KEYBOARD.SYS) file olvasása közben hiba.

SELECT. A jelzett file olvasása közben hiba történt. Feltehetően a használt lemez tönkrement. Próbálkozzunk a biztonsági másolattal.

READ ERROR IN: x:\xxxx\xxxx
- Olvasási hiba az x:\xxxx\xxxx file-on

EDLIN. Az x:\xxxx\xxxx file olvasása közben hiba történt. A file valószínűleg fizikai hibás.

REINSERT DISKETTE FOR DRIVE D AND STRIKE ENTER WHEN READY
- Helyezze vissza a lemezt az egységbe, majd nyomjon meg egy billentyűt

FORMAT. A /S paramétert adtuk meg, de a parancs nem tudta a rendszerfile-okat egyszerre beolvasni a memóriába, mert nem volt elég hely. Ezért újra kéri a rendszerlemezt. Helyezzük be, majd nyomjuk meg az < ENTER > billentyűt.

REPLACE < d:út\file név > (Y/N)?
- Helyettesítsem a < d:út\file név > file-t (I/N)?

REPLACE. A /P paramétert adtuk meg, s a parancs visszakérdez, hogy mely file-okat akarjuk helyettesíteni. Ha N-nel válaszolunk, akkor a file-t a parancs nem helyettesíti, ha Y-nal, akkor igen.

REPLACING < d:út\file név >
- A < d:út\file név > file-t helyettesíti

REPLACE. Tájékoztató üzenet, hogy a jelzett file helyettesítése megkezdődött.

REQUESTED LOGICAL DRIVE SIZE EXCEEDS THE MAXIMUM AVAILABLE SPACE
- a kívánt logikai egység méret meghaladja a maximális rendelkezésre álló méretet.

FDISK. Az általunk megadott logikai egység mérete túl nagy. A program válaszként mindig a használható legnagyobb értéket kínálja.

REQUESTED PARTITION SIZE EXCEEDS THE MAXIMUM AVAILABLE SPACE
- a kívánt partíció-méret meghaladja a maximálisan rendelkezésre állót.

FDISK. Az általunk megadott partíció-méret túl nagy. A program válaszként mindig a használható legnagyobb értéket kínálja.

REQUESTED STACK SIZE EXCEEDS 64K
(ANGEFORDETE STAPELGRÖßE ÜBERSCHREITET 64K)
- Az igényelt verem (STACK) mérete túllépi a 64 K-t.

LINK. Ha a STACK SIZE = üzenet jelenik meg, akkor 64 K-nál kisebb értéket kell megadni.

RELOCATION TABLE OWERFLOW
- A relokációs tábla túlcsordult

LINK. A szerkesztett program több mint 13000 távoli (LONG) attributumú vezérlésátadást és mutatót tartalmaz, ezért a LINK már nem képes ezek tárolására. Az ilyen hivatkozások számát csökkenteni kell.

RESIDENT PART OF PRINT INSTALLED
(RESIDENTER TEIL VON PRINT INSTALLIERT)
- A PRINT rezidens része installálva van

PRINT. Amikor a PRINT-et először használjuk, akkor jelenik meg ez az üzenet. Egy program a memóriába került, amely a következő PRINT parancsokat feldolgozza. Ezzel a szabad memória 3200 byte-tal csökkent.

RESIDENT PORTION OF MODE LOADED
(RESIDENTER TEIL VON MODE GELADEN)
- A MODE rezidens része be van töltve

MODE. Amikor a MODE-t először hívjuk be nem kijelzési funkcióra, akkor a MODE egy részét is rezidenssé kell tenni.

RESIDENT PORTION OF NLSFUNC LOADED
- az NLSFUNC rezidens része betöltve.

NLSFUNC. Információs üzenet: a program rezidens része a tárba töltődött.

RESTORE FILE SEQUENCE ERROR
- Visszatöltési file szám hiba

RESTORE. A file-t a parancs nem tudta visszatölteni, mert a mentett lemezeket nem megfelelő sorrendben adtuk.

***** RESTORING FILES FROM DRIVE Y *****
- Az Y egységről az X egységre a visszatöltés megkezdődött.

RESTORE. Tájékoztató üzenet, mely jelzi, hogy az y egységről az x egységre megkezdődött a file-ok visszatöltése.

S...

SAME DRIVE SPECIFIED MORE THAN ONCE
- ugyanarra az egységre többször hivatkoztunk.

FASTOPEN. A parancsban egy-egy lemezegység betűjele csak egyszer szerepelhet.

SECTOR SIZE ADJUSTED
- Szektor mérete módosítva

VDISK. A konfigurációs parancs módosította az általunk megadott szektorméretet. Nem hibaüzenet.

SECTOR SIZE TOO LARGE IN FILE <FILENAME>
(SEKTORGRÖBE IN DATEI <DATEINAME> ZU GROB)
- A szektor mérete túl nagy a <filenév> file-ban

START. Az operációs rendszer töltésekor a <filenév> periféria kezelőben a DOS számára definiáltnál nagyobb szektorméretet adtunk meg. A paramétert módosítani kell, majd újraindítani a rendszert.

SEGMENT LIMIT TOO HIGH, EXCEEDS 1024
- Túl nagy szegmensszámot adtunk meg, a maximum 1024

LINK. A /x paraméterben megadott szegmensszám nagyobb a megengedettnél. Módosítsuk, és szerkesszük újra a programot.

SEGMENT LIMIT TOO HIGH
- Túl nagy szegmensszámot adtunk meg

LINK. A /x paraméterben (kezdő értéke 256) megadott számú szegmens részére a LINK nem tud elegendő helyet foglalni a memória táblázataiban. Csökkentsük a /x paraméter értékét, vagy szabadítsunk fel további memóriaterületet (pl. a VDISK-et).

SEGMENT SIZE EXCEEDS 64K
(SEGMENTGRÖBE ÜBERSCHREITET 64K)
- A szegmens mérete túllépi a 64 K-t

LINK. Szerkesztés során egy szegmens mérete túllépi az 64 K-t. A legnagyobb címtartomány viszont csak 64 Kbyte.

SHARE ALREADY INSTALLED
- A file megosztási környezetet már installáltuk.

SHARE. A file megosztási környezetet csak egyszer lehet betölteni, újra megkíséreltük. Nem hibaüzenet.

SHARING VIOLATION
- File megosztási hiba

XCOPY. A cél vagy forrás file-ok valamelyikét egy másik processzo zárolta, ezért az XCOPY nem tudja írni vagy olvasni. Várjunk, míg a file felszabadul, s adjuk újra ki a parancsot.

SOURCE AND TARGET DRIVES ARE THE SAME
- A forrás és a cél lemez ugyanaz

BACKUP vagy RESTORE. A forrás és a cél lemezegységet azonosnak adtuk meg. Adjunk meg különböző egységeket és adjuk ki újra a parancsot.

SOURCE DOES NOT CONTAIN BACKUP FILES
- A forrás lemezen nincsenek mentett file-ok

RESTORE. A parancsban megjelölt forrás lemezen nincsenek mentett file-ok, ezért a parancsot nem lehet végrehajtani.

SOURCE / PATH REQUIRED
- A forrás-hoz vezető utat meg kell adni

REPLACE. A parancsban nem szerepel egyetlen forrás út sem, holott legalább egyet meg kell adni.

SPECIFIED COMMAND SEARCH DIRECTORY BAD
- A parancshoz vezető út hibás

Akármelyik parancs küldheti. A parancsot megelőző lemez és útleírás rossz. Kijavítása után ismételjük meg a parancsot.

SPECIFIED DRIVE NOT EXIST, OR IS NON-REMOVABLE
- A megadott lemezegység nem létezik, vagy fix

DISKCOPY vagy DISKCOMP. A parancsban megadott lemezegység nincs a rendszerben vagy fix lemez. Javítsuk ki a lemezegység azonosítóját és adjuk ki újra a parancsot.

SOURCE DISKETTE BAD OR INCOMPATIBLE
- A forrás lemez rossz vagy nem megfelelő formájú

DISKCOPY. A lemez olvasása közben hiba történt. Ennek oka lehet, hogy bizonyos szektorok fizikailag tönkrementek, de az is, hogy egy 360 Kbyte-os lemezegységbe egy nagy kapacitású lemezt helyeztünk.

STACK SIZE EXCEEDS 65535 BYTES
(STAPELGRÖBE ÜBERSCHREITET 65535 BYTES)
- A verem mérete túllépi a 65535 byte-ot

LINK. A megadott méretnek kisebbnek kell lennie mint 65535 byte.

SYMBOL DEFINED MORE THAN ONCE
(SYMBOL MEHRFACH DEFINIERT)
- Többszörösen definiált szimbólum

A LINK olyan modulokat talált, amelyek ugyanazt a szimbólumot definiálják.

SYMBOL TABLE OVERFLOW. SYMBOL TABLE CAPACITY EXCEEDED
(KAPAZITÄT DER SYMBOLTABELLE ÜBERSCHRITTEN)
- A szimbólum tábla méretét túlléptük

LINK. Túl sok és túl hosszú nevet adtunk meg. Méretük meghaladja a 256 Kbyte-ot. Rövidebb és kevesebb szimbólumot kell használni.

SYNTAX ERROR
(SYNTAXFEHLER)
- Szintaktikus hiba

DOS. A parancs formája helytelen. Ellenőrizni kell a parancs formáját.

SYSTEM FILES RESTORED. THE TARGET DISK MAY NOT BE BOOTABLE
- Rendszer file-ok visszatöltve. Lehet, hogy a lemez nem betölthető

RESTORE. A DOS rendszerfile-okat (IBMBIO.COM, IBMDOS.COM) visszatöltöttük. Előfordulhat, hogy a lemez nem lesz betölthető, mert korábbi DOS verziót írtunk rá.

SYSTEM TRANSFERED
- A rendszer átvitele megtörtént

FORMAT. Az üzenet a /S paraméter használatakor jelenik meg, jelezve, hogy a DOS rendszerfile-ok is rákerültek a lemezre.

SYSTEM WILL NOW RESTART. INSERT DOS DISKETTE IN DRIVE A:, PRESS ANY KEY WHEN READY
- A rendszer újraindul. Helyezze be a DOS lemezt az A: meghajtóba, majd nyomjon meg egy billentyűt

FDISK. Tájékoztató üzenet: az FDISK használata után újra kell indítani a rendszert, hogy az újjáírt partíciós táblát használja. Helyezzük be a DOS lemezt az A: egységbe, és nyomjunk meg egy tetszőleges billentyűt.

T...

TARGET CANNOT BE USED FOR BACKUP
- A cél lemez nem használható mentésre

BACKUP. A cél lemez használhatatlan. Ha cserélhető, próbáljuk meg egy másik lemezzel. Ha fix, akkor indítsuk újra a rendszert és adjuk ki újra a parancsot.

TARGET DISKETTE MAY BE UNUSABLE
(ZIELDISKETTE MÖGLICHERWEISE UNBRAUCHBAR)
- Lehetséges, hogy a cél lemez nem használható

DISKCOPY. Ez az üzenet egy olyan hibaüzenet után áll, amely egy nem ismételten fellépő írási vagy olvasási hibára utal. A céllemezt I/O hiba miatt már nem lehet majd másolni.

TARGET DISKETTE UNUSABLE
- A cél lemez használhatatlan

DISKCOPY. A másolás során a cél lemezen hibát talált a rendszer. Egy másik lemezről ismételjük meg a másolást.

TARGET DISKETTE WRITE PROTECTED. CORRECT, THEN STRIKE ANY KEY
(ZIELDISKETTE IST SCHREIBGESCHÜTZT, BERICHTIGEN UND EINE TASTE BETÄTIGEN)
- A cél lemez írásvédett. Javítsa ki, majd nyomjon le egy billentyűt

DISKCOPY. Írásvédett lemezre akartunk másolni.

TARGET IS FULL
- A cél lemez megtelt

RESTORE. Mentés közben a lemez megtelt. Próbáljuk meg a mentést egy másik lemezre, amin több szabad hely van.

TERMINATE BATCH JOB (Y/N)?
(STAPELJOB BEENDEN (J/N)?
- A kötegelt (BATCH) job befejeződjön (I/N)?

DOS. Ez az üzenet akkor jelenik meg, ha a <CTRL-BREAK> billentyűt nyomjuk le, miközben a DOS parancs file-t dolgoz fel. Ha a feldolgozást be akarjuk fejezni, akkor a Y-t, ill. a J-t kell lenyomni. Ha N-t adunk meg, akkor csak az a parancs fejeződik be, amely éppen feldolgozás alatt állt. A feldolgozás a következő paranccsal folytatódik.

TERMINATE BY USER
- A felhasználó megszakította a programot

LINK. A <CTRL-BREAK> billentyűvel megállítottuk a programot, amit már csak újraindítani lehet. Tájékoztató üzenet.

THE CURRENT ACTIVE KEYBOARD TABLE IS xx
WITH CODE PAGE: yyy
THE CURRENT ACTIVE CON CODE PAGE IS: zzz
- a jelenlegi aktív billentyűzet: xx
- a kód-lap: yyy
- a CON eszköz kód-lapja: zzz

KEYB. A parancs lekérdező funkciójának használatakor kapjuk a fentieket válaszul. Információs üzenet.

THE CURRENT ACTIVE PARTITON IS X
(DIE MOMENTANE AKTIVE PARTITION IST X)
- A pillanatnyilag aktív partició az x

FDISK. Az "aktív partíció változtatása" lehetőség kijelzi a fix lemez pillanatnyilag aktív partícióját.

THE LAST FILE WAS NOT RESTORED
(DIE LETZTE DATEI WURDE NICHT ZURÜCKGESPEICHERT)
- Az utolsó file visszatöltése nem történt meg

RESTORE. A visszatöltés befejeződött, mielőtt a megadott file-okat teljezen visszatöltötte volna, vagy kevés volt a hely a fix lemezen. A RESTORE a részben visszatöltött file-t törölte.

THE ONLY BOOTABLE PARTITION ON DRIVE 1 IS ALREADY MARKED ACTIVE
- az első fix lemez egyetlen tölthető partíciója már aktív.

FDISK. Információs üzenet.

THERE WAS/WERE xxx ERRORS DETECTED
(ES WURDE(N) ZAHL FEHLER ENTDECKT)
- A szerkesztés hibaszintje

LINK. A szerkesztés végén jelenik meg ez a tájékoztató üzenet. A szerkesztés sikerességéről, illetve sikertelenségéről kapunk információt.

TOO MANY BLOCKS DEVICES
- Túl sok blokkos periféria kezelő

CONFIG.SYS. A rendszer maximum 26 blokkos perifériakezelőt tartalmazhat. A konfigurációs file-ban ennél többet kíséreltünk meg implementálni. Csökkentsük ezek számát és indítsuk újra a rendszert.

TOO MANY DRIVE ENTRIES
- túl sok lemezegység megadása

FASTOPEN. A parancsban túl sok egységet soroltunk fel. Csökkentsük ezek számát, majd ismételjük meg a parancsot.

TOO MANY EXTERNAL SYMBOLS, LIMIT 510 PER MODULE. TOO MANY EXTERNAL SYMBOLS IN ONE MODULE
(ZU VIELE EXTERNE SYMBOLE IN EINEM MODUL)
- Túl sok külső hivatkozás egy modulban

LINK. Egy modulban legfeljebb 510 külső hivatkozás (external reference) lehet.

TOO MANY FILES OPEN
- Túl sok nyitott file

EDLIN. A parancs megkísérelte a megadott file-t megnyitni, de nem tudta. Növeljük meg a konfigurációs file-ban a FILES = értéket, és indítsuk újra a rendszert.

TOO MANY GROUP-, SEGMENT-, AND CLASS-NAMES, LIMIT 254 PER MODULE
- Túl sok csoport-, szegmens-, és osztály név, a maximum 254 modulonként

TOO MANY GROUPS
(ZU VIELE GRUPPEN)
- Túl sok csoport

LINK. A csoportok száma a DGROUP-pal együtt legfeljebb 10 lehet.

TOO MANY GRPDEFS, LIMIT 8 PER MODULE
- Túl sok GRPDEF, modulonként 8 lehet.

LINK. A szerkesztő valamelyik modulban több mint 8 GRPDEF utasítást talált. A számukat csökkenteni kell.

TOO MANY LIBRARIES SPECIFIED
TOO MANY LIBRARIES, LIMIT IS 16
(ZU VIELE BIBLIOTHEKEN ANGEGEBEN)
- Túl sok könyvtárat adtunk meg

LINK. Több mint 8 könyvtárat adtunk meg.

TOO MANY NAME ENTRIES
- túl sok név megadása

FASTOPEN. Összesen 999 file- és út-név adható meg. A feleslegeseket hagyjuk el és ismételjük meg a parancsokat.

TOO MANY OPEN FILES
- túl sok nyitott file

XCOPY. A rendszert úgy konfiguráltuk, hogy a parancs nem tud kellő file-kezelőt használni. A CONFIG.SYS file-ban a FILES paraméterét legalább 20-nak válasszuk meg és indítsuk újra a rendszert.

TOO MANY OVERLAYS
(ZU VIELE ÜBERLAGERUNGEN)
- Túl sok az overlay

LINK Az overlay mérete legfeljebb 65535 lehet.

TOO MANY PUBLIC SYMBOLS
(ZU VIELE ALLGEMEINE SYMBOLE)
- Túl sok közös szimbólum

LINK. A közös szimbólumok legfeljebb 1024 byte-ot foglalhatnak el.

TOO MANY SEGMENTS, LIMIT 255 PER MODULE
- Túl sok szegmens, a maximum 255 lehet modulonként

LINK. Az objekt program több mint 255 szegmenst tartalmaz, ami nem megengedett. Csökkentsük a szegmensek számát, és futtassuk újra a LINK-et.

TOO MANY SEGMENTS, USE /X:N (256 < N < 1025)
- Túl sok szegmens, használja a /X:N (256 < N < 11025) kulcsot

LINK. A lehetséges-nél több szegmens szerepel a szerkesztő programban. Az /X paraméter megadásával ill. a szegmensek számának csökkentésével szerkesszük újra a programot.

TOO MANY TYPEDEFS, LIMIT 255 PER MODULE
- Túl sok TYPEDEF, a maximum 255 lehet

LINK. A TYPEDEF rekordokat a programszerkesztő a közös (COMMON) változók leírására használja, egy modulban legfeljebb 255 lehet, s ezt a határt túlléptük.

TOP LEVEL PROCESS ABORTED, CANNOT CONTINUE
- A legfelső szintű processz abortált, nem lehet folytatni

A rendszer betöltése közben a COMMON.COM vagy egy másik parancs lemezhibát jelzett, s az ABORT-ot választottuk. Másik lemezzel kell a rendszert indítani.

TOTAL DISC SPACE IS XXXX CYLINDERS
(GESAMTER PLATTENBEREICH BETRÄGT XXXX ZYLINDER)
- A teljes lemezterület mérete xxxx cilinder

FDISK. A teljes fix lemezterületet kiírja

TRACK 0 BAD-DISC UNUSABLE
(SPUR 0 FEHLERHAFT - DISKETTE UNBRAUCHBAR)
- A 0. sáv a lemezen nem használható

FORMAT. A lemez 0. sávján a következők helyezkednek el: boot-rekord, tartalomjegyzék rekord, file-leíró táblázat (FAT). Valamelyik szektor hibás, a lemez használhatatlan.

TRANSFER SIZE ADJUSTED
- Az átvitel mérete módosítva

VDISK. Az egyszerre átvihető rekordok száma nem esett az 1-8 intervallumba, ezért a VDISK 8-ra állította be.

TREE PAST THIS POINT NOT PROCESSED
(VERZEICHNISSTRUKTUR AB DIESEM PUNKT NICHT VERARBEITET)
- A tartalomjegyzék struktúrát e ponttól nem tudja feldolgozni

CHKDSK. A CHKDSK az éppen átvizsgált elérési út feldolgozását nem tudja folytatni, mivel a 0. sáv hibás.

U...

UNABLE TO COPY KEYBOARD ROUTINE
- A billentyűzet rutint nem lehet másolni

A KEYBxx.COM file megnyitása és olvasása közben hiba történt. Ha ezt a file tárolása okozta, később próbálkozzunk újra.

UNABLE TO CREATE DIRECTORY
(INHALTSVERZEICHNIS KANN NICHT ERSTELLT WERDEN)
- Nem lehet a tartalomjegyzéket létrehozni

MKDIR. Az előállítandó altartalomjegyzék már létezik; vagy az elérési út, amelyet a felhasználó megadott, nem található; vagy a főtartalomjegyzék már megtelt, és nem lehet újabb bejegyzést hozzáfűzni.

UNABLE TO CREATE KEYB TABLE IN RESIDENT MEMORY
- a KEYB a táblázatait nem tudja elhelyezni a rendelkezésére álló helyen.

KEYB. A parancs rezidens részét már előzőleg a memóriába töltöttük. A most kiadott parancs táblázatai nem férnek el az előzőleg erre a célra foglalt helyen. Indítsuk újra a DOS-t és a KEYB-t a megfelelő konfigurációval töltsük a memóriába.

UNABLE TO WRITE BOOT
(EINLEITENDER LADASATZ KANN NICHT GESCHRIEBEN WERDEN)
- A boot írása nem lehetséges

FORMAT. A lemez első sávja vagy a DOS partíció hibás. A kezdeti betöltőt (boot) nem lehetett erre a sávra felírni. A lemez nem használható DOS rendszerlemezként.

UNABLE TO SHIFT SCREEN LEFT
- A képernyőt nem lehet balra tolni

MODE. A képernyő a megjelölt értékkel már nem tolható el balra.

UNABLE TO SHIFT SCREEN RIGHT
- A képernyőt nem lehet jobbra tolni.

MODE. A képernyő a megjelölt értékkel már nem tolható el jobbra.

UNABLE TO WRITE BOOT
- a boot rekordot nem lehet felírni.

FORMAT. A lemez első sávjának első szektora rossz. Dobjuk ki.

UNEXPECTED END-OF-FILE ON LIBRARY
(UNERWARTETES DATEIENDE IN DER BIBLIOTHEK)
- Nem várt EOF a könyvtárban

LINK. Ez általában a könyvtári állományban levő hibára vezethető vissza.

UNEXPECTED END-OF-FILE ON SCRATCH FILE
UNEXPECTED END OF FILE ON VM.TMP
(UNERWARTETES DATEIENDE IN VM.TMP)
- Nem várt EOF a VM.TMP-ben

LINK. Például a VM.TMP file-t tartalmazó lemezt kivettük.

UNRECOGNIZED COMMAND IN CONFIG.SYS
(NICHT ERKANNTER BEFEHL IN CONFIG.SYS)
- Nem felismerhető parancs a CONFIG.SYS-ben

START. Érvénytelen parancs található a CONFIG.SYS file-ban. A hibás parancsot ki kell javítani, majd a rendszert újra kell indítani.

UNRECOGNIZED SWITCH ERROR: XXXXXX
- Az xxxxxx kapcsoló rossz

LINK. A / jel után egy nem azonosítható xxxxxx kulcsot használtunk.

UNRECOVERABLE ERROR ON DIRECTORY
- A főtartalomjegyzék rossz

CHKDSK. A főtartalomjegyzék ellenőrzése közben a parancs hibát talált.

UNRECOVERABLE FILE SHARING ERROR
- Felismerhetetlen file megosztási hiba

Valamelyik parancs azonosíthatatlan file megosztási hibát észlelt.

UNRECOVERABLE FORMAT ERROR ON TARGET. DISKETTE UNUSABLE
(NICHT WIEDERHERSTELLBARER FORMATFEHLER AUF ZIELDISKETTE / ZIELDISKETTE UNBRAUCHBAR)
- Nem javítható formátum hiba a cél lemezen. A cél lemez használhatatlan.

DISKCOPY. Nem javítható hiba a cél lemez formázásánál. A lemez hibás adatokat tartalmaz.

UNRECOVERABLE READ ERROR ON DRIVE X TRACK XX SIDE X
(NICHT WIEDERHERSTELLBARER LESEFEHLER IN LAUFWER X SPUR XX, SEITE X)
- Nem javítható olvasási hiba az X meghajtón az XX sávon X oldalon

DISKCOMP. Az adatok olvasása a megadott meghajtóban lévő lemezen négy kísérlet során sem sikerült. A hiba helyét az üzenet megadja.

UNRECOVERABLE READ ERROR ON SOURCE TRACK XX, SIDE X
(NICHT WIEDERHERSTELLBARER LESEFEHLER AUF QUELLENDISKETTE SPUR XX, SEITE X)
- Nem javítható olvasási hiba a forráslemez x oldalán xx sávon

DISKCOPY. A forráslemezről négy sikertelen olvasás történt. A DISKCOPY folytatja az olvasást, de a másolat hiányos adatokat tartalmazhat.

UNRECOVERABLE VERIFY ERROR ON TARGET TRACK XX, SIDE X
(NICHT WIEDERHERSTELLBARER PRÜFFEHLER AUF ZIELDISKETTE SPUR XX, SEITE X)
- Nem javítható ellenőrzési hiba a cél lemez x oldalán, xx sávján

DISKCOPY. Négy sikertelen írás-ellenőrzés volt a cél lemezen, a DISKCOPY folytatja a másolást, de a másolat hiányos adatokat tartalmazhat.

UNRECOVERABLE WRITE ERROR ON TARGET TRACK XX, SIDE X
(NICHT WIEDERHERSTELLBARER SCHREIBFEHLER AUF ZIELDISKETTE SPUR XX, SEITE X)
- Nem javítható írás hiba a cél lemez x oldalán, xx sávján

DISKCOPY. Négy sikertelen írás volt a cél lemezre. A DISKCOPY folytatja a másolát, de a másolat hiányos adatokat tartalmazhat.

UNRESOLVED EXTERNALS: LIST
(UNBEKANNTE EXTERNE SYMBOLE: LISTE)
- Feloldhatatlan külső hivatkozás: lista

LINK. A kilistázott külső hivatkozások (external reference) definiálatlanok a modulokban. Az így előállított load-modult nem szabad végrehajtani.

V...

VDISK NOT INSTALLED - INSUFFICIENT MEMORY
- A memória lemez - memóriahiány miatt - nincs installálva

VDISK. A memória mérete nem megfelelő. Ennek oka lehet, hogy a paraméterek korrigálása után is 64 Kbyte-nál kisebb használható memória van, de lehet az is, hogy a kiterjesztett memória mérete nem megfelelő.

VDISK VERSION v.vv VIRTUAL DISK X
- VDISK verzió v.vv Memória lemez: x

VDISK. A memória lemez sikeres installálásáról tájékoztató üzenet. A memória lemezre az x betűvel hivatkohatunk.

VERIFY IS ON/OFF
- Ellenőrzés be/ki kapcsolva

VERIFY. Tájékoztató üzenet. ON = a rendszer ellenőrzi a lemezre történő írást, OFF = nem.

VM.TMP IS AN ILLEGAL FILENAME AND HAS BEEN IGNORED
(VM.TMP IST EIN UNZULASSIGER DATEINAME UND WURDE IGNORIERT)
- VM.TMP illegális file-azonosító, ezért figyelmen kívül marad

LINK. A VM.TMP-t tárgy (object) modul azonosítóként nem lehet megadni.

VOLUME LABEL (11 CHARACTERS, ENTER FOR NONE)?
(VOLUME-NAME (11 ZEICHEN, KEINEE NAME- > EINGABE)?)
- A kötetazonosító (11 karakter, ha nem kell, ENTER)?

FORMAT. Ez az üzenet a kötetazonosítót kéri, ami legfeljebb 11 karakter lehet. Ha a felhasználó nem akar kötetazonosítót felírni, akkor csak az ENTER billentyűt kell megnyomnia.

W...

WARNING! ALL DATA ON NON-REMOVABLE DISK DRIVE X WILL BE LOST PROCEED FORMAT (Y/N)?
- Figyelem! Az x fix lemez összes adata elvész. Megformázzam (I/N)?

FORMAT. Fix lemezegység formázása törli az összes adatot. Ha Y-nal válaszolunk, akkor a formázás megtörténik, ha N-nel, akkor nem.

WARNING! ALL DATA IN THE DOS PARTITON WILL BE DESTROYED. DO YOU WISH TO CONTINUE...? [d]
(ACHTUNG! ALLE DATEN IN DER DOS PARTITON WERDEN ZERSTÖRT. SOLL FORTGEFAHREN WERDEN...? [d])
- Figyelem! Minden adat a DOS partícióban elvész. Akarja folytatni...? [d]

FDISK. A "DOS partíció törlés" lehetőség választása esetén jelenik meg ez az üzenet. A DOS partíció valamenyi adata törlésre kerül. Ha csak az < ENTER > billentyűt nyomjuk le, akkor nem törlődnek, ha viszont az Y-t, ill. a J-t és üzenetvég-et is adunk, akkor törlődnek.

WARNING! DATA IN THE EXTENDED DOS PARTITION COULD BE DESTROYED. DO YOU WISH TO CONTINUE?
- Vigyázat: a kibővített DOS partíción levő adatok megsemmisülhetnek. Akarja folytatni?

FDISK. A kibővített DOS partíció törlésekor kapjuk a fenti üzenetet. Ha ennek ellenére törölni akarjuk a partíciót, akkor az < 4 > billentyű után nyomjuk meg az < ENTER > -t!

WARNING! DATA IN THE PRIMATY DOS PARTITION COULD BE DESTROYED. DO YOU WISH TO CONTINUE?
- Vigyázat: az elsődleges DOS partícióban levő adatok megsemmisülhetnek. Akarja folytatni?

FDISK. Az elsődleges DOS partíció törlésekor kapjuk a fenti üzenetet. Ha ennek ellenére törölni akarjuk a partíciót, akkor az <Y> billentyű után (Yes) nyomjuk meg az <ENTER>-t is!

WARNING! DIRECTORY FULL
xxx FILE(S) RECOVERED
(WARNUNG! INHALTSWERZEICHNIS VOLL)
- Figyelem! A tartalomjegyzék megtelt

RECOVER. Az újraelőállítandó file-ok számára nincs hely a tartalomjegyzékben. Néhány file-t egy üres lemezre kell másolni, majd az eredetiről törölni. Ezután újra végrehajtható a RECOVER.

WARNING! DISKETTE IS OUT OF SEQUENCE. REPLACE THE DISKETTE, OR CONTINUE. STRIKE ANY KEY WHEN READY
(WARNUNG! DISKETTE NICHT FOLGERICHTIG. DISKETTE AUSTAUSCHEN. WENN BEREIT, EINE TASTE BETÄTIGEN)
- Figyelem! Lemez-sorrend hiba, cserélje a lemezt, vagy nyomjon le egy tetszőleges billentyűt

RESTORE. A tartalék (backup) lemez nem a következő a sorban. A lemezt le kell cserélni. Ha a felhasználó biztos abban, hogy az átlépett (kihagyott sorszámú) lemezen nincs olyan file, amit elő kell állítani, akkor a RESTORE egy tetszőleges billentyű lenyomásával folytatható. Ez az üzenet állandóan ismetlődik, ha a felhasználó olyan lemezt akar átlépni, amelyen az újra előállítandó file egy része található.

WARNING! FILE XX IS A READ-ONLY FILE. REPLACE THE FILE (Y/N)?
(WARNUNG! DATEI XX IST EINE NUR-LESE-DATEI. DATEI AUSTAUSCHEN (J/N)?)
- Figyelem! Az xx file csak olvasható. Cserélje a file-t (I/N)?

RESTORE. A megadott file csak olvasható. Ha ki akarjuk cserélni, akkor Y-t, ill. J-t, különben N-t kell megadnunk. Az <ENTER> lenyomásával folytatható a művelet. Ezt az üzenetet csak a /P paraméter megadása esetén kapjuk.

WARNING! FILE XX WAS CHANGED AFTER IT WAS BACKED UP. REPLACE THE FILE (Y/N)?
(WARNUNG! DATEI XX WURDE NACH DER SICHERUNG GEÄNDERT. DATEI AUSTAUSCHEN (J/N)?)
- Figyelem! Az xx file a mentés óta megváltozott. Az állományt lecseréljük (I/N)?

RESTORE. A megadott file a fix lemezen más dátumot és időt tartalmaz, mint a tartalék lemezen található kimentett file. Az Y, ill. J válasz magadásával a file-t lecseréljük. A

RESTORE az <ENTER> lenyomása után folytatódik. Ezt az üzenetet a /P paraméter megadása esetén kapjuk.

WARNING! FILES IN THE TARGET DRIVE d:\ ROOT DIRECTORY WILL BE ERASED. STRIKE ANY KEY WHEN READY
- Figyelem! A d: egység főtartalomjegyzékében levő valamennyi file elvész. Ha rendben, nyomjon le egy billentyűt

BACKUP. Figyelmeztetés: a parancs a főtartalomjegyzékben levő valamennyi file-t törölni fogja. A végrehajtás akkor kezdődik, ha megnyomunk egy tetszőleges billentyűt.

WARNING! NO FILES WERE FOUND TO BACK UP
(ACHTUNG! KEINE DATEIEN ZUR SICHERUNG GEFUNDEN)
- Figyelem! A mentésre kijelölt file nem található.

BACKUP. A mentésre kijelölt file nem található a fix lemezen.

WARNING! NO FILES WERE FOUND TO RESTORE
(ACHTUNG! KEINE DATEIN ZUM ZURÜCKSPEICHERN GEFUNDEN)
- Figyelem! A visszatöltésre kijelölt file nem található

RESTORE. A lemezen nem található a file, amelyet a fix lemezre vissza akarunk tölteni.

WARNING! NO STACK SEGMENT
(ACHTUNG! KEIN STAPELSEGMENT)
- Figyelem! Nincs verem szegmens

LINK. A megadott tárgy modulok egyike sem tartalmaz hivatkozást a verem (stack) szegmensre. (Ha .COM-ot akarunk generálni a programból, ennek az üzenetnek szerkesztéskor meg kell jelennie.)

WARNING! READ ERROR ON EXE FILE
(ACHTUNG! LESEFHELER IN EXE-DATEI)
- Figyelem! Olvasási hiba az .EXE kiterjesztésű file feldolgozásakor

EXE2BIN. A file olvasása során hiba jelentkezett. Az EXE2BIN megpróbálja a továbblépést, de az eredmény file bizonyos esetekben használhatatlanná válik. Próbálja a file-t másik meghajtóról olvasni.

WARNING! TARGET IS FULL
- Figyelem! A cél lemez megtelt

BACKUP vagy RESTORE. A cél lemez megtelt. Nem lehet további file-okat feldolgozni.

WARNING! THE PARTITION MARKED ACTIVE ISNOT BOOTABLE
- Vigyázat: az aktív partíció nem tölthető.

FDISK. A program figyelmeztet: az aktív partíció nem tölthető. Ha ez így marad, akkor a fix lemezről nem tudjuk a rendszert elindítani. Vélhetően nem ez volt a szándékunk.

WRITING xxx BYTES
- xxx byte kiírása megkezdődött

DEBUG. Tájékoztató üzenet, hány byte-ot ír ki a program.

X...

X IS NOT A CHOISE. ENTER A CHOISE.
(X IST KEINE AUSWAHL. EINGABE EINER AUSWAHL)
- Az x nem választható. Egy másik lehetőséget adjon meg.

X IS NOT CHOISE. ENTER Y OR N
(X IST KEINE AUSWAHL. EINGABE VON J ODER N)
- Az x nem választható. Megadható válasz Y, ill. J, vagy N.

XXX FILES ADDED
- xxx file-t adott hozzá

REPLACE. Tájékoztató üzenet: a parancs a cél lemezhez xxx file-t adott hozzá.

XXXXXX IS NOT A VALID LIBRARY
- xxxxxx rossz könyvtár

LINK. A könyvtár file megadása rossz.

XXXXXXXXXX BYTES DISC SPACE FREED
(XXXXXXXXXX BYTES PLATTENSPEICHER FREIGEMACHT)
- xxxxxxxxx byte lemez terület szabaddá téve

CHKDSK. Az előzőleg kijelölt terület nem került lefoglalásra. Ez most felszabadult.

xxxxxxxx CODE PAGE DRIVER CANNOT BE INITIALIZED
- az xxxxxxxx kód-lap kezelőt nem lehet inicializálni.

PRINTER.SYS vagy DISPLAY.SYS. A DEVICE konfigurációs parancsban rossz paramétereket adtunk meg. Ellenőrizzük a CONFIG.SYS file-t, majd indítsuk újra a rendszert.

xxxxxxxx DEVICE DRIVER CANNOT BE INITIALIZED
- az xxxxxxxx periféria-kezelőt nem lehet inicializálni.

DOS. A DEVICE konfigurációsparancsban az xxxxxxxx kezelőre vonatkozó parancs rossz. Ellenőrizzük a CONFIG.SYS file-t, majd indítsuk újra a rendszert.

XXXX ERROR ON FILE YYYY
(XXXX FEHLER IN DATEI YYYY)
- xxxx hiba az yyyy file-ban

PRINT. Ez az üzenet a nyomtatón jelenik meg. Miközben yyyy file-ból a nyomtatás

számára adatokat olvastunk , egy xxxx típusú lemezhiba lépett fel. A file kiírása befejeződött.

XXX LOST CLUSTERS FOUND IN YYY CHAINS
(XXX VERLORENE BEREICHE GEFUNDEN IN YYY BEREICHEN)
- xxx elveszett csoport található yyy darab láncban

CHKDSK. Az xxx csoportok elveszett blokkjait megtalálta a CHKDSK. A CHKDSK lehetővé teszi azt, hogy ezeket tötöljük, vagy önálló file-okként előállítsuk.

XXXXXX OF YYYYYY BYTES RECOVERED
- yyyyyy byte-ból xxxxxx-et visszaállított

RECOVER. Tájékoztató, hogy összesen hány byte-nyi információt sikerült visszaállítani, s hogy mennyiből.

***** BACKING UP FILES TO DISKETTE XX *****
(*** SICHERUNG VON DATEIEN AUF DISKETTE XX***)
- *** A file-ok mentese xx lemezre történt

BACKUP. Ezt az üzenetet a kimentett file-ok listája követi az adott lemezről.

***** FILES WERE BACKED UP XX/XX/XXXX *****
(*** DATEIN WURDEN GESICHERT XX/XX/XXXX ***)
- *** A file mentése történt xx/xx/xxxx***

RESTORE. A megadott időpontban történt a mentés. (A dátum formája attól függ, hogy a CONFIG.SYS file-ban a nemzet-specifikus paramétert beállítottuk-e.)

***** RESTORING FILES FROM DISKETTE XX *****
(***ZURÜCKSPEICHERN VON DATEIEN VON DISKETTE XX ***)
- *** A file-ok visszatöltése xx lemezről történik ***

RESTORE. Ezt követi az ismét előállított (visszatöltött) file-ok listája az adott lemezről.

nnn VERSION OF GRAPHIC CHARACTER SET TABLE IS ALREADY LOADED
- a grafikus karakterkészlet nnn verziója már a memóriában van.

GRAFTABL. A paranccsal egy már a memóriában levő és még használhatókód-lapot akartunk betölteni. Információs üzenet. nnn lehetséges értékei: 437, 860, 863 és 865.

nnn VERSION OF GRAPHIC CHARACTER SET TABLE IS NOW LOADED
- a grafikus karakterkészlet nnn verziója betöltve.

GRAFTABL. Információs üzenet.

-MORE-
(-FORTSETZUNG-)
- Folytatás

MORE. A képernyő megtelt, de további adatok várnak kijelzésre. A következő képernyőtöltés egy tetszőleges billentyű lenyomásával kezdeményezhető.

10 MISMATCES - ENDING COMPARE
(10 VERGLEISCHFEHLER - VERGLEICHEN BEENDET)
- 10 összehasonlítási hiba - befejezve az összehasonlítás

COMP. Már 10 nem egyező byte van az összehasonlítandó file-okban. A további összehasonlításnak nincs értelme.

B. Függelék

Parancsok, alparancsok csoportosított betűrendes listája

Parancsok	Funkció
APPEND	Bejárási utak kijelölése tetszőleges kiterjesztésű file-okra.
ASSIGN	A periféria-hozzárendelések megváltoztatására használható, például floppy-meghajtó esetén.
ATTRIB	A file-ok archiválási és csak-olvasható jelzőinek ki- ill. bekapcsolása.
BACKUP	A fix lemezen levő file-ok (esetleg a teljes fix lemez tartalmának) kimentése.
BREAK	A BREAK billentyű figyelésének beállítása, állapotának lekérdezése.
CHCP	Kód-lap váltás valamennyi eszközre.
CHDIR	Az aktuális tartalomjegyzék változtatása. *Rövidített formája*: **CD.**
CHKDSK	A lemez állapotáról ad információt. Hiba esetén a fizikai tárolásnak megfelelően javítja a file-leíró táblázatot (FAT).
CLS	A képernyő törlése.
COMMAND	Második parancsfeldolgozó processzor elindítása.
COMP	File-ok összehasonlítása. Az összehasonlításra kerülő file-oknak azonos méretűnek kell lenni. Legfeljebb tíz eltérést jelez ki.
COPY	File-ok másolása, duplikálása, összekapcsolása. Bináris és szöveges (text) állományok is összefűzhetők.
CTTY	Alternatív konzol kijelölése.
DATE	A rendszer-dátum lekérdezése, beállítása.
DEL	File-ok törlése a tartalomjegyzékből.
DIR	A file-ok tartalomjegyzék bejegyzésének listázása. Teljes és tömörített formában is használható.
DISKCOMP	Floppy lemezek összehasonlítása (fizikai, szektoronként).
DISKCOPY	Floppy lemezek fizikai másolása.
ERASE	File-ok törlése a tartalomjegyzékből. *Rövidített formája*: **DEL.**
EXE2BIN	A végrehajtható .EXE formából .COM formájú programot készít.
EXIT	Kilépés a második COMMAND-ból.
FASTOPEN	File-ok megtalálásának lerövidítése.
FDISK	Fix lemezegység partíciójának kijelölése.
FIND	Megadott karaktersorozat keresése szöveges file-ban.
FORMAT	Lemezek formázása, inicializálása, a DOS rezidens file-jainak elhelyezése.
GRAFTABL	A grafikus jelkészlet betöltése a memóriába.
GRAPHICS	A grafikus nyomtatás lehetőségének beállítása (színes-grafikus adapter szükséges).
JOIN	Két fizikailag különböző lemezegység összekapcsolása egy logikai egységgé.

Parancsok	Funkció
KEYB	A DOS 3.30-as verziójának billentyűzet-kezelő programja.
KEYBxx	Az adott ország nemzeti billentyűzetét kezelő rutin betöltése.
LABEL	Lemez kötetazonosítójának (címkéjének) megadása.
MKDIR	A fastruktúrájú tartalomjegyzékben altartalomjegyzék létrehozása. *Rövidített formája*: **MD**.
MODE	A képernyő, nyomtató üzemmódjának beállítása, és az adatátviteli vonalak inicializálása.
MORE	File-ok képernyő méretű oldalakra tagolása.
NLSFUNC	Kiterjesztett ország-függő információk és bizonyos kód-lap kezelő (pl. CHCP) eljárások engedélyezése.
PATH	A bejárási utak kijelölése a fastruktúrájú tartalomjegyzékben. (Csak .EXE, .COM és .BAT kiterjesztés jó!)Környezet paraméter is (SET parancs).
PRINT	A nyomtatás várakozási sorának vezérlése (háttér nyomtatás).
PROMPT	A PROMPT karaktersor beállítása.
RECOVER	File-ok ismételt előállítása. A sérült lemezterületen kívül, vagy a tartalomjegyzék meghibásodása esetén új bejegyzések elkészítése a fileleíró táblázat (FAT) alapján.
RENAME	File-ok átnevezése. *Rövidített formája*: **REN**.
REPLACE	A cél tartalomjegyzék file-jainak helyettesítése a forrás file-okkal. (Új file nem keletkezik a cél lemezen).
RESTORE	A fix lemez állományainak visszatöltése a tartalék (backup) lemezekről.
RMDIR	Üres altartalomjegyzék törlése. *Rövidített formája*: **RD**.
SELECT	DOS installálása egy új (hajlékony) lemezen, adott billentyűzet rutinnal és dátum formával.
SET	A program-környezetben paraméterek elhelyezése, illetve a környezet-paraméterek lekérdezése.
SHARE	File megosztási környezet installálása.
SORT	Szöveges file-ok rendezése (növekvő vagy csökkenő sorrendben a rekordok megadott pozíciójától).
SUBST	Adott útra vagy lemezegységre másik lemezegységként hivatkozhatunk.
SYS	A DOS rezidens file-jainak elhelyezése egy lemezen.
TIME	A rendszer-idő beállítása, lekérdezése.
TREE	Tartalomjegyzék fastruktúrájának kilistázása.
TYPE	File-ok tartalmának kiíratása (szöveges/text típus esetén).
VER	A DOS verziójának lekérdezése.
VERIFY	A "lemezre írás-ellenőrzés" kapcsoló beállítása, állapotának lekérdezése.
VOL	Lemez kötetazonosítójának lekérdezése.
XCOPY	File-ok vagy teljes altartalomjegyzék másolása egy másik lemezre vagy altartalomjegyzékbe.

Batch alparancsok	Funkció
CALL	Parancs-file-ból másik parancs-file meghívása.
ECHO	A parancsok kijelzésének be-, illetve kikapcsolása.
FOR	DOS parancsok többszöri végrehajtása.
GOTO	A feldolgozás a címkével megjelölt sortól folytatódik.
IF	Feltételes parancs-végrehajtás.
PAUSE	A parancsok feldolgozásának felfüggesztése.
REM	Megjegyzések (REMarks) elhelyezése a parancs-kötegben.
SHIFT	A parancs-paraméterek léptetése (több, mint tíz paraméter esetén).

Konfigurációs parancsok	Funkciók
BREAK	A BREAK billentyű figyelésének beállítása.
BUFFERS	Az I/O műveletekhez használt lemez pufferek száma.
COUNTRY	A használt dátum és idő kijelzési formátum megadása.
DEVICE	A DOS által használandó perifériakezelők megadása.
FCBS	Az egyszerre megnyitható FCB-k (file controll blokk) száma.
FILES	Az egyidőben megnyitott file-ok maximális száma.
LASTDRIVE	A maximálisan használható lemezegység betűjele.
SHELL	A felhasználói parancs-processzor azonosítója.
STACKS	A hardware megszakítások kezeléséhez szükséges veremméret megadása.

C. Függelék

Parancsok, alparancsok formája

Parancs	Típus	Forma
APPEND	K	**APPEND** *d:út*[;[*d:*]*út*...]
		APPEND [/X][/E]
		APPEND [;]
ASSIGN	K	**ASSIGN** [*x*=*y*[...]]
ATTRIB	K	**ATTRIB** [+R][-R][+A][-A][*d:*][*út*]*file-név*[.*kit*][/S]
BACKUP	K	**BACKUP** [*d:*][*út*][*file-név*[.*kit*]][*d:*][/S][/M][/A][D:*hh-nn-éé*]
BACKUP	K	**BACKUP** [*d:*][*út*] [*d:*][/S][/M][/A][/P]
(MS DOS)		[/D:*dátum*][/T:*idő*][/L:*file-név*[.*kit*]]
BACKUP	K	**BACKUP** [*d:*][*út*][*file-név*[.*kit*]][*d:*][/S][/M][/A]
(3.30)		[/D:*dátum*][/T:*idő*][/F][/L[:[*d:*][*út*]*file-név*[.*kit*]]]
BREAK	B	**BREAK** [ON|OFF]
CHCP	K	**CHCP** [*kl*]
CHDIR	B	**CHDIR** [*d:*][*út*]
		CD [*d:*][*út*]
CHKDSK	K	**CHKDSK** [*d:*][*út*][*file-név*[.*kit*]][/V][/F]
CLS	B	**CLS**
COMMAND	K	**COMMAND** [*d:*][*út*][/P][/C *karaktersorozat*][/E:*xxxxx*]
COMP	K	**COMP** [*d:*][*út*]*file-név*[.*kit*][*d:*][*út*][*file-név*[.*kit*]]
COPY	B	**COPY** [/A][/B][*d:*][*út*]*filenév*[.*kit*][/A][/B]
		[*d:*][*út*][*file-név*[.*kit*]][/A][/B][/V]
		COPY [/A][/B][*d:*][*út*]*filenév*[.*kit*][/A][/B]
		[+[*d:*][*út*]*file-név*[.*kit*][/A][/B]...]
		[[*d:*][*út*]*file-név*[.*kit*][/A][/B][/V]
CTTY	B	**CTTY** *egységnév*
DATE	B	**DATE** [*hh-nn-éé*] | [*éé-hh-nn*] | [*nn-hh-éé*]
DEL	B	**DEL** [*d:*][*út*]*file-név*[.*kit*]
DIR	B	**DIR** [*d:*][*út*][*filenév*[.*kit*]] [/P][/W]
DISKCOMP	K	**DISKCOMP** *d: d:*[/1][/8]
DISKCOPY	K	**DISKCOPY** *d: d:*[/1]
ERASE	B	**ERASE** [*d:*][*út*]*file-név*[.*kit*]
EXE2BIN	K	**EXE2BIN** [*d:*][*út*]*file-név*[.*kit*][*d:*][*út*][*file-név*[.*kit*]]
EXIT	B	**EXIT**
FASTOPEN	K	**FASTOPEN** *d:*[*nnn*]...
FDISK	K	**FDISK**
FIND	K	**FIND** [/V][/C][/N] *karaktersorozat*[[*d:*][*út*]*file-név*[.*kit*]...]

FORMAT	K	**FORMAT** [*d:*][/S][/1][/8][/N][/B][/V][/N:*xx*][/T:*yy*]
GRAFTABL	K	**GRAFTABL** [437 | 860 | 863 | 865 | /STATUS]
GRAPHICS	K	**GRAPHICS** [*nyomtató-típus*][/R][/B][/LCD]
JOIN	K	**JOIN** [*d1: d2*:[*út*]] | [*d:*/D]
KEYB	K	**KEYB** [*xx*[,[*yyy*],[[*d:*][*út*]*file-név*[.*kit*]]]]
KEYBxx	K	**KEYBxx**
LABEL	K	**LABEL** [*d:*][*címke*]
MKDIR	B	**MKDIR** [*d:*]*út*
		MD [*d:*]*út*
MODE	K	**MODE LPT#:**[*n*][,[*m*][,P]]
		MODE *n*
		MODE [*n*],*m*[,T]
		MODE COM*n:baud*[,*paritás*[,*adatbitek*[,*stopbitek*[,P]]]]
		MODE *egység* **CODEPAGE PREPARE =** ((*kl*)[*d:*][*út*]*file-név*[.*kit*])
		MODE *egység* **CODEPAGE PREPARE =** ((*kl-lista*)[*d:*][*út*]*file-név*[.*kit*])
		MODE *egység* **CODEPAGE SELECT =** *kl*
		MODE *egység* **CODEPAGE** [/STATUS]
		MODE *egység* **CODEPAGE REFRESH**
MORE	K	**MORE <** [*d:*][*út*]*file-név*[.*kit*]
NLSFUNC	K	**NLSFUNC** [[*d:*][*út*]*file-név*[.*kit*]
PATH	B	**PATH** [*d:*]*út*[;[*d:*]*út*...]
		PATH [;]
PRINT	K	**PRINT** [/D: *egységnév*][/B: *pufferméret*][/U: *foglalt*][/M: *nyomtat*] [/S:*várakozik*][/Q: *sorméret*][/C][/T][/P][*d:*][*út*][*file-név*.[*kit*]...]
PROMPT	K	**PROMPT** *prompt jelek*

$ - $ karakter
t - rendszer idő
d - rendszer dátum
p - aktuális tartalomjegyzék
v - a DOS verziója
n - aktuális meghajtó
g - > karakter
l - < karakter
b - ' karakter
q - = karakter
h - kurzor visszaléptetése, megelőző karakter törlése
e - ESCape karakter
_ - CR/LF a képernyő következő sorának elejére kerül a kurzor

RECOVER	K	**RECOVER** [*d:*][*út*]*file-név*[.*kit*]
		RECOVER [*d:*]

RENAME	B	**RENAME** [*d:*][*út*]*file-név*[*.kit*]*file-név*[*.kit*]*file-név*[*.kit*]
		REN [*d:*][*út*]*file-név*[*.kit*]
REPLACE	K	**REPLACE** [*d:*][*út*]*filenév*[*.kit*][*d:*][*út*][/A][/P][/R][/S][/W]
RESTORE	K	**RESTORE** *d:* [*d:*][*út*]*filenév*[*.kit*][/S][/P]
RESTORE	K	**RESTORE** *d:* [*d:*][*út*]*file-név*[*.kit*][/S][/P]
(3.30)		[/B:*dátum*][/A:*dátum*][/M][/N][/L:*idő*][/E:*idő*]
RMDIR	B	**RMDIR** [*d:*]*út*
		RD [*d:*]*út*
SELECT	K	**SELECT** [[A:|B:] *d:*[*út*]] *xxx yy*
SET	B	**SET** [*név* = [*paraméter*]]
SHARE	K	**SHARE** [/F:*fileméret*][/L:*zárások*]
SORT	K	**SORT** [/R][/+*n*]
SUBST	K	**SUBST** [*d: d1:út*] | [*d:*/D]
SYS	K	**SYS** *d:*
TIME	B	**TIME** [*óó:pp:mm.xx*]
TREE	K	**TREE** [*d:*][/F]
TYPE	B	**TYPE** [*d:*][*út*]*filenév*[*.kit*]
VER	B	**VER**
VERIFY	B	**VERIFY** [ON|OFF]
VOL	B	**VOL** [*d:*]
XCOPY	K	**XCOPY** [*d:*][*út*][*filenév*[*.kit*]][*d:*][*út*][*filenév*[*.kit*]]
		[/A][/D][/E][/M][/P][/S][/V][/W]

Batch alparancs	Típus	Forma
CALL	B	**CALL** [*d:*][*út*]*file-név*
ECHO	B	**ECHO** [ON|OFF|*üzenet*]
FOR	B	**FOR** *%%változó* **IN**(*értékek halmaza*) **DO** *parancs*
GOTO	B	**GOTO**[:]*címke*
IF	B	**IF** [NOT] *feltétel parancs* A feltétel alakja: **ERRORLEVEL** *szám* *sztring 1* = = *sztring 2* **EXIST** [*d:*][*út*]*file-név*[*.kit*]
PAUSE	B	**PAUSE** [*üzenet*]
REM	B	**REM** [*megjegyzés*]
SHIFT	B	**SHIFT**

Konfigurációs parancsok	**Forma**
BREAK	**BREAK = ON | OFF**
BUFFERS	**BUFFERS** = *xx* (1< =xx< =99)
COUNTRY	**COUNTRY** = *xxx*
COUNTRY	**COUNTRY** = *xxx*,[*yyy*]
(3.30)	**COUNTRY** = *xxx*,[*yyy*],[*d:*][*út*]*file-név*[*.kit*]
DEVICE	**DEVICE** = [*d:*][*út*]*filenév*[*.kit*][*paraméterek*]
	DEVICE = DRIVER.SYS /D:*ddd*[**/T:***ttt*][**/S:***sss*][**/H:***hh*][**/C**][**/W**][**/F:***f*]
	DEVICE = VDISK.SYS [*bbb*][*sss*][*ddd*][**/E**[*:m*]]
	DEVICE = ANSI.SYS
	DEVICE = DISPLAY.SYS CON[**:**] = (*típus* [,[*hkl*][,*n*]])
	DEVICE = DISPLAY.SYS CON[**:**] = (*típus* [,[*hkl*][,(*n*,*m*)]])
	DEVICE = PRINTER.SYS LPT#[**:**] = (*típus* [,[*hkl*][,*n*]])
	DEVICE = PRINTER.SYS LPT#[**:**] = (*típus* [,[(*hkl1*,*hkl2*,...)][,*n*]])
FCBS	**FCBS** = *m*,*n*
FILES	**FILES** = *xx* (1< =xx< =99)
LASTDRIVE	**LASTDRIVE** = *x*
SHELL	**SHELL** = [*d:*][*út*]*filenév*[*.kit*][*paraméter1*][*paraméter2*]
STACKS	**STACKS** = *n*,*s*

D. Függelék

File-név kiterjesztések

A PC DOS által használt file-név kiterjesztések és rövid magyarázatuk:

Kiterjesztés	Magyarázat
.BAK	a file korábbi változata
.BAS	BASIC forrásprogram
.BAT	parancs-file (batch file)
.COM	végrehajtható (gépi kódú) program közvetlenül betölthető formában
.CPI	kód-lapokra vonatkozó információt tartalmazó file
.EXE	végrehajtható (gépi kódú) program áthelyezhető formában
.HEX	hexadecimális karaktereket tartalmazó file
.LIB	programkönyvtár (library)
.MAP	memóriatérkép a globális szimbólumokról
.OBJ	tárgy (objekt) program
.SYS	periféria-kezelő vagy annak tartozéka
.TMP	ideiglenes munka file
.$$$	ideiglenes munka file

E. függelék

Eszköznevek listája

A DOS parancsokban, ahol a file-specifikációt kell írnunk, az alábbi eszköznevek használata megengedett (Az eszköznevek utáni kettőspont használata nem kötelező.):

Eszköznév	Input vagy output	Magyarázat
AUX:	input és output	az első aszinkron adapter
COM1:	input és output	az első aszinkron adapter
COM2:	input és output	a második aszinkron adapter
COM3:	input és output	a harmadik aszinkron adapter (DOS 3.30-tól)
COM4:	input és output	a negyedik aszinkron adapter (DOS 3.30-tól)
CON:	input és output	input esetén billentyűzet, output esetén a képernyő
LPT1:	output	az első párhuzamos nyomtató
LPT2:	output	a második párhuzamos nyomtató
LPT3:	output	a harmadik párhuzamos nyomtató
NUL:	input és output	nemlétező (dummy) eszköz. Olvasásakor file vége jelet kapunk, írásakor tényleges átvitel nem történik
PRN:	output	az első párhuzamos nyomtató

%%, 113
%, 113
.BAT, 105
.COM, 18
.EXE, 18
<Ctrl-PtrSc>, 32
<F1>, <F2>...<F10>, 29
<F6>, 25
aktuális blokk, 245
aktuális lemezegység, 38
aktuális meghajtó, 34
aktuális prompt, 202
aktuális rekord, 244
aktuális tartalomjegyzék, 34, 61
alparancs, 105
altartalomjegyzék, 60, 220
ANSI.SYS, 173, 174
append, 139
APPEND, 79
ASCII rendezési súly, 188
ASCII érték, 188
ASCII-kód, 41
assemble, 151
ASSIGN, 94
ATTRIB, 54
attributum byte, 246
attributum, 46, 55
AUTOEXEC.BAT, 23, 107
auxiliary carry, 146
AX, 143, 146
BACKUP, 85, 111
Basic I/O System, 236
batch üzemmód, 12
batch-file, 105
befejezési kód, 110
BIOS paraméter blokk, 215, 228
BIOS, 14, 17, 236
blokkos berendezések, 212
boot record, 17
boot szektor, 214
boot, 204, 205, 214
BP, 143, 146, 205
BP:SI, 256
Break, 171
BREAK, 95
breakpoint, 154
BUFFERS, 172
BX, 143, 146
CALL, 114
carry, 146
CHCP (kódlap váltás), 193
CHDIR, 34
CHDIR, 61
CHKDSK, 75
cilinder, 82
CLS, 92
command processor, 65
COMMAND, 90
COMMAND.COM, 18, 202
COMP, 48
COMPARE, 48, 149
CONFIG.SYS, 171
COPY, 41, 131
COUNTRY, 172
CS, 143, 146, 205
Ctrl-Break, 252
CTTY, 190
CX, 143, 146
DATE, 92
DEBUG, 27, 141
delete, 125
Device Header Control Blokk, 254
device header, 223
DEVICE, 173
DF, 143
DI, 146, 205
DIR, 70
direction, 146
directory, 11, 12, 220
disassemblálás, 141, 153
Disk Transfer Area, 143
DISKCOMP, 74
DISKCOPY, 24, 72
DISPLAY.SYS, 143
DISPLAY.SYS-kódlap, 177
DOS, 11
driver, 223
DRIVER.SYS, 173, 178
DS, 143, 146, 205
DSALLOCATION, 169
DTA, 279, 287
dump, 145

duplaszó, 245
DX, 143, 146
ECHO, 108
EDLIN, 27, 115
egy pont (.), 71
egység leíró tábla, 228
elérési út, 35
elsődleges DOS partíció, 81
elsődleges és másodlagos file-azonosító, 49
end, 128
enter, 148
environment, 236
ERASE, 51
ERRORLEVEL, 110
ES, 143, 146, 205
EXE2BIN, 191
EXIST, 110
EXIT, 91
FASTOPEN, 59
FAT (File Allocation Table), 21
FCB, 273
FCB-orientált eljárás, 243
FCBS, 181
FDISK, 25, 81
file, 11
file allocation table (FAT), 75
file attributum, 245
file átnevezés, 52
file azonosító, 35, 36
File Control Block, 156
file kezelés, 243, 245
file kiterjesztés, 222
file kontroll blokk (FCB), 181
file leíró blokk, 243
file leíró tábla, 214, 217
file megosztási környezet, 55
file méret, 248, 250
file-név kiterjesztés, 71
file-név, 17, 35, 248
file-pointer, 251
file-szám, 243, 251
file-vezérlő blokk, 156
file-vég jel, 49
FILES, 182
fill, 147
FIND, 186
fix lemez, 85
flag, 143
FOR, 112
FORMAT, 26, 65
formattál, 65
főtartalomjegyzék, 33, 39
funkció, 251, 260
funkcióhívás, 211, 243
futtatható file, 161
függvényhívás, 205
globális file, 11
globális file-név, 36, 42
go, 154
GOTO, 112
GRAFTABL, 102
GRAPHICS, 103
gyökér tartalomjegyzék, 33
handler, 228
hexadecimális, 160
hibakeresés, 11
HIGH, 167, 169
IBMBIO.COM, 17, 52, 65
IBMBIO.SYS, 202
IBMDOS.COM, 17, 52, 65
IBMDOS.SYS, 202
IF, 110
inicializálás, 228
input, 160
insert, 119
INT, 209
interaktív üzemmód, 162
interrupt vektor, 102
interrupt vektortábla, 210, 281
interrupt, 146, 252
IP, 143, 146, 205
JOIN, 78
karakteres berendezés, 212, 260
karakteres periféria, 231
karaktersorozat, 132
KEYBGR, 25
KEYBxx, 101
két pont (..), 71
kiegészítő DOS partíció, 81
kilépési cím, 252
kilépési rutin, 253
kiterjesztés, 35, 244

konfigurációs file, 171
konfigurációs parancsok, 171
könyvtár-file, 162
kötegelt feldolgozás, 11
kötetazonosító, 65
LABEL, 72
LASTDRIVE, 183
lemez-file, 167
lemezegység, 86
lemezkezelés, 213
lemezátviteli terület, 246
LINE, 170
LINK, 15, 161
list, 120
load, 157
logikai egység, 82
logikai rekordméret, 245
logikai sorszám, 213
MAP, 167
map-file, 163, 166
megszakítás, 207, 252, 281
megszakítási pont, 154, 207
megszakítási rendszer, 205
meleg újraindítás, 203
memóriablokk, 241
memória leíró blokk (MCB), 241
memórialemez, 183
MKDIR, 60
MODE, 96
MORE, 186
move, 129, 150
name, 156
NLSFUNC, 193
old, 170
opció, 161
ország-függő információk, 193
overflow, 146
page, 122
paragrafus, 241
parancs-file, 12, 105
parancs-processzor, 18, 183
parancsblokk, 225
parity, 146
partíció, 82, 214
partíciós tábla, 214
PATH, 63
PAUSE, 109
PC, 143
periféria, 260
piping, 12
PRINT, 57
PRINTER.SYS, 173, 179
proceed, 155
programkörnyezet, 193, 236
programszegmens prefix (PSP), 143, 237
prompt, 143
PROMPT, 189
PSP (Program Segmens Prefix), 193, 236
quit, 128, 151
random, 248, 251, 279
random írás, 247, 279
random olvasás, 247, 279
RECOVER, 53
register, 146, 205
rejtett (hidden) file, 52, 246
rekord abszolút sorszáma, 247
relatív cím, 144
REM, 108
RENAME, 52
replace, 135
REPLACE, 47
RESTORE, 88
rezidens program, 241, 258
RMDIR, 62
sáv (track), 18, 213
search, 132, 150
SELECT, 68
SET, 193
SHARE, 55
SHELL, 183
SHIFT, 113
SI, 143, 146, 205
sorszám, 120
SORT, 187
SP, 143, 146
spooling, 57
SS, 143, 146, 207
STACK, 168, 183
standard I/O átirányítása, 184
SUBST, 77
SYS, 69
százalék, 106

szegmenscím, 153, 241
szegmensregiszter, 143, 206
szegmesszám, 170
szektor, 213
szekvenciális, 246, 251, 276
szó, 244
szöveg-file, 49
tartalomjegyzék, 11, 12, 35, 220
tárgy-file, 161
TIME, 93
trace, 155
track, 18
transfer, 137
TREE, 63
TYPE, 56
unassemble, 153
válasz-file, 164
VDISK.SYS, 173, 180
VER, 94
veremhasználat, 183
veremmutató, 143
VERIFY, 77
VOL, 71
volume label, 26
warm reset, 205
winchester, 213
write, 138, 158
XCOPY, 45

IRODALOMJEGYZÉK

Ashley, R., Fernandez, J. N.:
PC DOS: Using the IBM PC Operating System
J. Wiley & Sons, Inc., 1983.

Bradley, D. J.:
Assemblly Language Programming for the IBM Personal Computer
Prentice-Hall, 1984.

Disk Operating System
(Version 2.00, 3.00, 3.10, 3.20, 3.30)
(IBM PC Computer Language Series)
IBM Corp., 1984-1987.

Graham, L. J., Field, T.:
Your IBM PC
A Guide to the IBM PC (DOS 2.00) and XT
ed.: Osborne/McGraw-Hill, Berkeley, California

IBM Personal Computer XT
Technical reference
IBM Corp., 1983.

IBM Personal Computer AT
Technical reference
IBM Corp., 1984.

IBM DOS Technical Reference
(Version 3.00, 3.10, 3.20, 3.30)
IBM Corp., 1984-1987.

King, R. A.:
The IBM PC DOS Handbook
Berkeley, California SYBEX, Inc., 1983.

Microsoft Macro Assembler for the MS-DOS Operating System
Reference Manual
Microsoft Corporation 1984, 1985.

Microsoft
MS-DOS Operating System
Programmer's reference Manual
Microsoft Corp., 1984-1987.

Morse, S. P.:
The 8086/8088 Primer
Hayden Book Co., 1982.

Norton, P.:
Inside the IBM PC
Robert J. Brady Co., 1983.

Norton, P.:
MS-DOS and PC-DOS User's Guide
Robert J. Brady Co., 1984.

Reector, R. - Alexy, G.:
The 8086 Book
Osborne/McGraw-Hill, 1980.

Sargent, M. III - Shoemaker, R. L.:
The IBM Personal Computer From the Inside Out
Addison-Wesley Publishing Co., 1984.

Szilassy Bertalan:
Az i8086 mikroprocesszor 1. kötet Utasításkészlet
LSI ATSz, Bp., 1983.

SzKI SCI-L Proper 16/A professzionális személyi számítógép
Operációs rendszer kézikönyv
SzKI SCI-L, Bp., 1983.

SzKI SCI-L PROPOS-16
Operációs rendszer és a WINCHESTER DISK használata
Felhasználói leírás
SzKI SCI-L, Bp., 1984.

Top Wiew
Programmer's Toolkit
IBM Corp., 1984.

EltSECRET

EltSECRET adatvédelmi rendszer

Termékünk, az EltSECRET, a SW'88 díjazott programjának, az EltGUARD-nak adatvédelemre továbbfejlesztett változata.

IBM PC-n működő adatvédelmi rendszer, mely nemcsak az adatokhoz és programokhoz való hozzáférést szabályozza, hanem adattitkosításra (kódolásra) is lehetőséget ad.

Az EltSECRET rendszer jellemzői:

- 64 hierarchia-szint a fájlok védelmére
- 127 alkönyvtárszint
- 2500 jogosult felhasználó
- 3 különböző bonyolultságú kódolási eljárás
- együttműködés lokális hálózati rendszerekkel
- felhasználói menürendszerek kialakítása
- kiegészítő hardverrel fokozott biztonság

Hardverkiegészítő használata esetén a felhasználók fizikai személyi azonosítót kapnak, jelszavuk mellett ez is kell a bejelentkezéshez. A herdver letiltja a floppyról történő rendszerbetöltést is.

A teljes program ára: **76.000,– Ft** (+ ÁFA).
A DEMO változat ára: **8.000,– Ft** (+ ÁFA).

Egy telephely összes számítógépéhez elegendő egy példány EltSECRET rendszert vásárolni.

Bővebb felvilágosítással, bemutatóval állunk rendelkezésére:

eltsoft

EltSOFT
Számítástechnikai és Reklámszervező GMK
1122 Budapest, Városmajor u. 7–9.
Telefon: 550-962 1536 Budapest, Pf. 390.

ELŐJEGYZÉST VESZÜNK FEL MONITOROK SZÁLLÍTÁSÁRA

MONOCHROME:

— 12 inches képátló
— TTL DIN bemenet

SZÍNES:

— 14 inches képátló
— TTL RGB bemenet
— összetett videójel-bemenet
— EUROCONNECTOR

RÖVID SZÁLLÍTÁSI HATÁRIDŐ FOLYAMATOS SZÁLLÍTÁS!

1 ÉV GARANCIA PHILIPS képcső

SZÁMSZÖV®
SZÁMÍTÁSTECHNIKAI KISSZÖVETKEZET

Címünk: Budapest XI., Hunyadi János utca 162.
Levélcím: 1430 Budapest, Postafiók 16. Telefon: 665-322.